A Statistical Analysis of the Aggregation of Crowdsourced Labels

by

David Szepesvari

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Crowdsourcing, due to its inexpensive and timely nature, has become a popular method of collecting data that is difficult for computers to generate. We focus on using this method of human computation to gather labels for classification tasks, to be used for machine learning. However, data gathered this way may be of varying quality, ranging from spam to perfect. We aim to maintain the cost-effective property of crowdsourcing, while also obtaining quality results. Towards a solution, we have multiple workers label the same problem instance, aggregating the responses into one label afterwards. We study what aggregation method to use, and what guarantees we can provide on its estimates. Different crowdsourcing models call for different techniques – we outline and organize various directions taken in the literature, and focus on the Dawid-Skene model. In this setting each instance has a true label, workers are independent, and the performance of each individual is assumed to be uniform over all instances, in the sense that she has an inherent skill that governs the probability with which she labels correctly. Her skill is unknown to us. Aggregation methods aim to find the true label of each task based solely on the labels the workers reported. We measure the performance of these methods by the probability with which the estimates they output match the true label. In practice, a popular procedure is to run the EM algorithm to find estimates of the skills and labels. However, this method is not directly guaranteed to perform well in our measure. We collect and evaluate theoretical results that bound the error of various aggregation methods, including specific variants of EM. Finally, we prove a guarantee on the error suffered by the maximum likelihood estimator, the global optima of the function that EM aims to numerically optimize.

Acknowledgements

First and foremost I would like to thank my supervisor Shai Ben-David for his patience, support and the freedom he has given me in the topic of my study. I appreciate his deep understanding of many subjects, the clarity of his thoughts and communication—compelling me to work towards the same—and the feedback he provided.

I am also grateful to my parents for the example they have set for me in all aspects of life; additionally, my father for making his invaluable knowledge, insights and experience available to me in general and in the context of this work, too; my mother, as well as brother-in-law Deon and friend Shreya for the motivation, time and attention they have granted me and this project.

Finally, I would like to acknowledge my friends, in particular Neeraj, and everyone else who made my time at the University of Waterloo as pleasant as it was.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Crowdsourcing, a recent buzzword, refers to the process of distributing jobs to a large group of people over the internet. The so-called workers get exposed to more opportunities through which they can earn a wage, and for the employers it provides a convenient, quick, and often cheap way of getting tasks done that would otherwise require hiring someone. Nonetheless, crowdsourcing is not always monetarily motivated; the Zooniverse[1] project is one of the numerous "volunteer research" platforms.

Our main motivation of using crowdsourcing is to collect labels for supervised learning. In modern-day techniques, predictors get trained on thousands, if not millions of examples. With crowdsourcing these labels can often be collected in a cheap and timely fashion; the caveat being that the quality of these labels is not guaranteed and in fact, a large part of it can even be spam. One solution is to request multiple workers to solve, or in our case label each problem. For many crowdsourcing problems, identifying the best solutions can be done manually. However, given the large quantity of data we are considering, there is a need for an automated method. We aim to aggregate responses, from a number of sources of varying reliability, into labels for each supervised learning instance that we believe are perfect with high confidence.

This is a popular and well-studied problem. We devote the next chapter to outlining and organizing a number of approaches taken in the literature. In the remainder of the thesis, we focus on the theoretical aspect of the problem, that is, we wish to see and provide guarantees on the performance of various aggregation methods. Moreover, we want these guarantees to come in the form of a mislabelling error: assuming that each problem to be labelled has a true label associated to it, we wish to obtain that label with high probability.

[1] https://www.zooniverse.org/

We select the one-coin model (an instance of the Dawid-Skene model) — which is simple, yet effective — and discuss in-depth the available knowledge pertaining to it. The key assumption in this model is that each worker providing labels has an inherent level of reliability that is uniform over her labelling, governing the probability with which her labels are correct. In order to better reflect on the performance of aggregation methods used and analysed in this setting we use a tabular representation, contrasting the guarantees offered for each, in various stylized scenarios.

In the model of our choice, one of the most common practical aggregation algorithms is the well known EM meta-algorithm. As EM is not directly guaranteed to perform well according to our natural measure, there have been various studies to establish the low mislabelling error (Gao and Zhou, 2013; Zhang et al., 2014). We uncover an important mistake in a proof given by Gao et al.Gao and Zhou (2013) pertaining to the performance of the global maximizer of the function EM optimizes. Furthermore, we set out to prove an upper bound in a similar setting. To be precise, we consider the maximum likelihood estimate and work towards a proof of an upper bound on its probability of mislabelling. We finish with a finite sample bound that controls the error rate of these estimates, but relies on a simple analytic statement that is well-justified, albeit not yet proven.

Outline. In Chapter 2, we detail the *crowdsourcing* scenario considered along with the common problems related to it. Chapter Chapter 3 specifies the model we work with in full mathematical detail while introducing some notation used throughout the thesis. We also quote and discuss relevant statements from the literature, and present their guarantees for an overview. Chapter 4 elaborates common techniques that will be used in the proofs towards the main result. Finally, Chapter 5 shows our main results leading up to a bound on the error rate of the maximum likelihood estimates. In Chapter 6, we conclude the thesis, reflecting on our result, its place in the literature, and outlining directions in which the investigation could and should continue. Chapter 7 holds the proofs of the mathematical claims made throughout the thesis.

Chapter 2

Crowdsourcing Problems

Our motivation is to collect labelled examples for (semi-)supervised learning. Historically there was a shift from supervised to semi-supervised learning motivated by the fact that in many domains unlabelled data is abundant, does aid learning, and labels are costly in comparison. With crowdsourcing, however, one now has access to cheap labelled data – though of possibly low quality. The goal is to have the cake and eat it too; can we boost the quality of the labels acquired from crowds while keeping the cost low?

In order to boost the label quality, we ask multiple workers – who are assumed to be labelling independently of each other – to label the same instance. We can report the majority label as our estimate with increased confidence. Clearly, then, a strategy is to keep adding such independent workers. However, we would like to keep costs as low as possible, therefore study whether there are more "efficient" ways to combine, or *aggregate* the labels provided by the workers. We will make the key assumption that the performance of each worker is in some way consistent over all instances of a task. Details vary between different works, but in general the inclusion of this assumption does enable significant improvements over the above mentioned majority voting in the label quality-cost trade-off.

For an illustration, consider the situation that there are some instances of the problem where all the workers specified the same label. This increases our confidence in these workers' abilities, and we trust them more than others. This, in turn, may reveal that there are other instances where workers appearing skilful are mostly in agreement. Again, we use this information to adjust our evaluation of the workers. Eventually, we get a sense of the workers' skill, while providing estimates for the labels that takes this into account. Although the pattern exploited in this illustration may be more subtle in typical data gathered through crowd-sourcing, it is still possible to utilize it.

Note that this process of gathering labels of unknown quality is not exclusive to crowdsourcing. The sources of the labels are irrelevant, as long as it is sensible to make the required assumptions. An important example is the aggregation of labels from already trained classifiers. The problem of aggregating classifiers is well studied in statistical learning, though traditionally this is done using labelled data. The crowdsourcing problem may be interpreted as the aggregation of classifiers using unlabelled data; the recent work of Jaffe et al. (2015) is, in fact, phrased in such a way.

This chapter presents the general framework we consider and provides an overview of the research studying it. We describe a typical crowdsourcing platform, as well as the nature of the data collected with it. We then identify three problem families dealing with crowdsourced data. Independently of these, we categorize works based on the relationship they assume between the labels to be recovered and those provided by workers.

2.1 The Nature of Crowdsourced Data

Since our study is motivated by the use of crowdsourcing to acquire labels for machine learning tasks, in particular classification, we often require a copious number of instances to be labelled by the crowd, with each individual instance quickly categorized by a human in exchange for a small sum of money on the scale of 1-3 cents. Such crowdsourcing tasks are often referred to as microtasks. We outline here key facts about Amazon Turk (Amazon, 2015), a crowdsourcing site, which was the first well-known such platform and is still popular in the microtasking community. Its system is representative of a large body of crowdsourcing sites:

- It is the workers that decide which tasks they will do. That is, employers cannot request workers, let alone workers of given accuracy. However there are ways – widely used in the hope of reducing the amount of spam – to filter the workers that can even attempt the task based on a simplistic system-wide measure of reported reliability. This does not actually seem to stop spammers, however (Ipeirotis, 2010).
- It is possible to require training of workers before they can attempt a task.
- It is possible to reject work submitted by a worker, meaning they will not be paid. This should not be done lightly, though, as it may result in a bad reputation of the employer, with workers not even attempting future tasks (Chen, 2012; Kumar, 2014).

- It is reasonable to require workers complete some given number of instances per payout. However, more often each payout is tied to a small number of instances with the system making it simple to continue with new instances of the same task. In fact, workers tend to grind on one task if they like it. This means we can mostly expect two types of workers: ones who complete a few tasks, and ones who do many –measured in 10s, 100s or even 1000s (Chen, 2012; Kumar, 2014; redditer, 2014).
- Each user's ID is available with their submission, so they can be identified.
- It is possible to specify that workers cannot repeat the same instance multiple times.

Note that the last bullet point aids in actually enforcing the assumption that sources are independent. Furthermore, that the fact that IDs are available with each submission allows us to identify and track each worker, which is needed in order to utilize the assumption on the workers' labelling performance. On the other hand, it is apparent that controlling the quality of responses is difficult; it is expected to see spammers whose responses carry no information about the true label of the instances. Workers exerting effort may also make varying numbers of mistakes.

The Amazon Turk framework has a serious limitation: it is not possible to request specific workers. Put differently, workers cannot be allocated to specific tasks or even instances. This would likely increase the efficiency of a smart microtasking and label aggregation system – though it also brings up various design questions. We concentrate on the system offered by Amazon Turk and other similar platforms, as it is the most widespread contemporary design.

A word on label quality: It is important to note that in practice the way labelling problems are presented to the worker have a tremendous effect on the quality of each individual worker label. While not the prime example of a microtask, in a Zooniverse task (Zooniverse, 2015) involving labelling animals on pictures taken by deployed cameras in the Serengeti, the user interface helps with identifying the type of animal. This is based on a number of features that individually become simple labelling tasks (e.g. the shape of the horn of the animal, if any) and act as filters on all possible answers to the original instance. In effect, the daunting task is broken down into a sequence of easily manageable problems. Similarly, even attributes of the batch of instances submitted to a crowdsourcing system, such as the number of total available tasks, the default payment delay and the payout also influence the quantity and the quality of the labels from the workers (Kumar, 2014; Chen, 2012). While it may be interesting to study how to motivate or optimize for quality labels, it is not our focus. We take the labels received granted, and aim to aggregate them effectively.

2.2 Mathematical Formulations of Label Aggregation

There are various directions in studying label aggregation; we characterize them in two independent ways. We first describe a range of aggregation problems with increasing degree of freedom, then we categorize works by how they model the relationship between the true and worker labels. First, however, let us establish a common dictionary, and with it a precise statement of the scenario investigated.

Setup. In the crowdsourcing setting we have *instances* of *tasks*, each with its own *(true) label*. These labels, however, are generally not known to us, and we ask *workers* or *labellers* to label them. We see the labels they provide, and refer to these as *worker labels*. The workers are assumed to have an inherent *skill* that, vaguely speaking, captures how good they are at labelling instances of tasks. A *task* here refers to a type of labelling problem, containing many instances that are homogeneous in nature. We may speak of multiple tasks to express that the skill of a worker on one task may be different from her skill on another task. Unfortunately the term *task difficulty*, which should really be *instance difficulty*, corresponds to varying levels of labelling hardness amongst otherwise similar instances, even for the same worker. The reason for this terminology is that in the literature concerned solely with homogeneous instances each instance is, in fact, often referred to as a task. As the main focus of this thesis is the homogeneous setting, we will also adopt this convention starting in the next chapter. Until then we maintain the distinction for clarity. All instances are uploaded to a crowdsourcing system by *employers* or *requesters*. Requesters may use *gold(en) units*, also known as *gold standard*s, instances whose true label they know – these can be used to test workers. Finally, variables such as skills, task difficulties and sometimes the true labels as well, are often referred to as *parameters*.

Our goal is to recover the true label of each instance. We quantify the success of various methods as the fraction of instances whose label they recovered correctly. In the context of the statistical models used in the literature, we may say we *infer* or *estimate* the true labels. When analysing inference methods in this setting theoretically, our measure of success translates into the expected value of the fraction of mistakes made, or more generally the cumulative distribution of the same (random) quantity.

2.2.1 Categorization by Degrees of Freedom

We consider three types of problems:

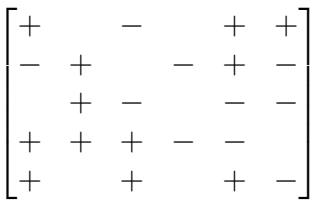

Figure 2.1: An example of a worker label matrix, the input to the Inference Problem. Here 5 workers (rows) provided labels for a total of 6 instances (columns) of a binary task. The entries in the matrix are the labels provided by the workers; entry (i, j) is the label assigned to instance j by worker i. If an entry is missing, the worker did not label the corresponding instance.

- Prediction Problem: given the parameters (other than the true labels), upon observing labels for a new instance output an estimate for its true label. More generally, we may only have estimates of the parameters (Berend and Kontorovich, 2014).
- Inference Problem: upon seeing only the labels provided by the workers on a number of instances, infer the true task labels, possibly other parameters as well. The input to the inference algorithm can be thought of as an incomplete matrix that collects the responses from the workers; see Figure 2.1 for an example. Additionally, the methods may incorporate golden data as well, though the key is that they effectively utilise the worker labels arriving from dubious sources in the estimation of the true label. As outlined above, we measure performance by the portion of true labels recovered. This is the classical crowdsourcing aggregation problem, with works including that of Whitehill et al. (2009); Welinder et al. (2010); Liu et al. (2012); Gao and Zhou (2013); Zhang et al. (2014); Jaffe et al. (2015) and this thesis.
- Instance Allocation: (interactively) allocate instances to workers in order to infer the true task labels (and other parameters) with as few queries as possible (Kolobov et al., 2013; Bragg et al., 2014; Abbasi-Yadkori et al., 2015).

The main results in Karger et al. (2011b, 2013, 2014) fix a specific allocation before seeing any labels, and solve the inference problem once the labels do arrive; this places them closer to the second category.

There are also various other directions studied that are outside the scope of this document; we highlight two here that are more closely related to our interests, though. Liu et al. (2015) consider the inference problem, but insert gold units after the data was collected, one by one, in order to try to get the best increase in label quality. A line of work in task allocation (Ho et al., 2013; Jung, 2014) considers allocating different tasks to workers,

t\r	-	+
-	0.8	0.2
+	0.3	0.7

Figure 2.2: A possible confusion matrix.
Here "t" stands for the "true label", "r" stands for the "response", and the real number in entry (i, j) is the probability that the worker will assign label j to an instance with true label i.

not simply instances. They aim to automatically find similar instances and utilize this information in the allocation methods.

2.2.2 Models of Labelling

Some sort of model describing the relationship between the true and the worker label is required for any result. Most papers in this domain explicitly work with a simple expression for the probability of a worker mislabelling an instance. We may phrase many such popular models in either the Dawid-Skene or the Joint CrowdSourcing (JoCR) model. We now describe these in detail, mentioning their variants that appear in the literature. Afterwards, some models that work under significantly different assumptions are also mentioned.

Dawid-Skene model. This model was introduced by Dawid and Skene in 1979 (Dawid et al., 1979) and has become a classical label crowdsourcing model forming the basis for a fruitful line of research (Karger et al., 2011b; Liu et al., 2012; Li et al., 2013; Berend and Kontorovich, 2014), recently culminating in methods claiming (near-)optimal results (Gao and Zhou, 2013; Karger et al., 2014; Zhang et al., 2014). Most work is set in the Inference Problem scenario.
We assume each worker has a latent confusion matrix describing the joint probability distribution over true and reported labels. An example of such a confusion matrix for the case of binary classification is shown in Figure 2.2. The worker with the confusion matrix shown on this figure will label negative instances correctly with probability 0.8, positive instances with 0.7. Other workers may have completely different confusion matrices.

Considerable amount of research is focused on labelling binary tasks; then the model is also referred to as the two-coin model. We may further specialize it by restricting the diagonal entries to be equal. This model is called the one-coin, or sometimes the Symmetric Dawid-Skene model. It models uniform behaviour over the two classes, and hence we lose the ability to model worker bias.

Joint CrowdSourcing (JoCR). The general JoCR framework was introduced by Kolobov et al. (2013). The main merit of JoCR is that it is able to model a possibly ever-changing pool of workers, allocating tasks to them in real time. Note that some of the work referenced in this category works only on a static pool, though, or was published outside of the JoCR framework but can be naturally rephrased in its notation.

There is a set of tasks Q, each task $q \in Q$ with a small number of possible answers $\mathcal{A}_q$. Each task q has an inherent difficulty d_q, and each worker $w \in \mathcal{W}$ has an inherent skill γ_w, where $\mathcal{W}$ is a pool of workers, which may or may not be changing from timestep to timestep.

The probability that worker w will label question q correctly is $p(d_q, \gamma_w)$, that is, it depends only on these two parameters. Note that this assumption, again, implies we cannot model bias towards certain labels. Different functions p may appear in different works, however, we often can re-parametrise one into the other. We present some examples that model truly different assumptions.

- The model that appeared most prominently, for example in the work of Bragg et al. (2014), is
$$p(d_q, \gamma_w) = \frac{1}{2}\left[1 + (1 - d_q)^{\frac{1}{\gamma_w}}\right],$$
where $d_q \in [0, 1]$ and $\gamma_w \in (0, \infty)$. Note that this assumption cannot model workers who tend to label incorrectly.

- For anonymous workers the model reduces to $\mathbb{P}(\text{correct label for } j) = \text{diff}_j$, that is, we only have difficulties. Here Abbasi-Yadkori et al. (2015) is interested in optimizing the order in which the instances of unknown difficulty should be put up for labelling.

- The model used by Whitehill et al. (2009) for inference also falls into this category:
$$p(d_q, \gamma_w) = \frac{1}{1 + e^{-d_q \gamma_w}}.$$
Here the probability of worker w labelling an instance with difficulty d_q correctly is modelled by the logistic function with input $-d_q\gamma_w$ for $\gamma_w \in [-\infty, +\infty]$ and $d_q \in [0, \infty]$. This function p generalizes the one-coin model by introducing difficulties.

We illustrate the existence of models that do not explicitly work with the probability of labelling correctly, rather they describe the process as a sequence of steps. Welinder et al. (2010), for example, describe:

1. Each instance is mapped to a vector characterization that correspond to features a perfect labeller would consider in order to label the instance. This step can capture some instances being more difficult than others.

2. Each individual worker observes these features with varying degrees of accuracy. This is modelled by using a gaussian noise of different variances on the different features, according to the skill of the worker.

3. The worker reports a label based on their internal affine decision boundary.

They explicitly incorporate biases, and model different levels of uncertainty quite well.

Chapter 3

Inference in the Dawid-Skene Model

We work in a special case of the Dawid-Skene model. Nonetheless, we first precisely introduce the model in its generality, then restrict our attention to its simplification. This will clarify the relationship between our and the general model, and will also allow us to discuss results more effectively. Though we have already outlined and provided context for the inference problem and the Dawid-Skene model, this chapter is devoted to the detailed and mathematically precise treatment of these topics, followed by an in-depth investigation of the theoretical literature on crowdsourcing in this setting.

3.1 Problem Definition

We follow a number of conventions. First, matrices and random variables will be denoted by capital letters unless otherwise specified. Vectors and scalars most often are lower case. We use the notation $[n] = \{1, 2, \ldots, n\}$, and $\mathbb{I}\{\cdot\}$ for the indicator function. The model will include a number of parameters, and these will be estimated. Generally the true parameters receive a star, as in p_i^* or p_*, estimates a hat $\hat{p}_i$. Note that these estimates are truly the output of estimators on the random data, and, as such, random variables. Arguments to functions playing roles similar to these parameters will stand without any decoration. We use $\mathbb{1}$ to denote vectors whose components are all one. All vectors are column vectors unless otherwise noted. We use $\cdot^\top$ to denote the transpose operation. Finally, we note that log denotes the natural logarithm.

General Dawid-Skene. There are t -for task- instances of a K-way classification task, with instances commonly indexed by j. Here we assume the labels are from the set $[K]$.

t\r	-	+
-	p_i^*	$1 - p_i^*$
+	$1 - p_i^*$	p_i^*

Figure 3.1: Confusion matrix for the one-coin model.
Here "t" stands for the "true label", "r" stands for the "response". p_i^* is the probability of returning the correct label, which is the same for both positive and negative labels in this model.

Each instance j has a true label $y_j^* \in [K]$, collectively denoted by $y_* \in [K]^t$. There are w workers, indexed by i, each with an inherent confusion matrix $P_i^* = (p_{i,cl}^*)_{c,l} \in [0,1]^{K \times K}$, $p_{i,cl}^*$ holding the probability that worker i labels an instance with true label c as l. That is, we must have $\sum_{l \in [K]} p_{i,cl}^* = 1$ for any $c \in [K]$. Here we assume that every worker labels every instance, though it is common to model missing values by assuming that whether a worker labels a task is like a Bernoulli random variable with some parameter specific to that worker. The worker responses may be formatted into a matrix. We denote the random variable corresponding to the reported label of worker i on instance j by Y_{ij}, and the random matrix that collects these by Y. The Dawid-Skene assumption, then, can be written as

$$\begin{aligned} \mathbb{P}\left(Y_{ij} = l\right) &= p_{i,y_j^* l}^*, && i \in [w], j \in [t], l \in [K] \text{ and} \\ \left(Y_{uv}\right)_{(u,v) \in A} &\perp \left(Y_{xy}\right)_{(x,y) \in B} && \text{for } A, B \subset [w] \times [t], A \cap B = \emptyset\,, \end{aligned}$$

where the second line records that the Y_{ij} are mutually independent.

One-coin model. As mentioned earlier, the one-coin model is the special case of the Dawid-Skene model where we only consider binary tasks, and assume that workers exhibit no bias toward either class. The simpler notation we introduce here is what the rest of the thesis will use.

In this case we redefine the labels to be either positive or negative, that is, $y_* \in \{\pm 1\}^t$. Due to the additional assumption on the confusion matrix, we now have just one parameter $p_i \in [0,1]$ determining the whole confusion matrix shown on Figure 3.1. Formally, for any worker i

$$p_{i,cl}^* = \begin{cases} p_i^*\,, & c = l\,; \\ 1 - p_i^*\,, & c \neq l, \end{cases}$$

that is, workers label correctly with probability p_i. Instead of working with p, we introduce $s = 2p - 1 \in [-1, +1]$, which communicates the nature of the worker better. The sign of

s indicates whether the worker is biased towards the correct label, or they are more likely reporting the wrong answer. The magnitude of s, in turn, clearly communicates how strong the worker's bias is. In particular, if $s = 0$, we have a spammer; if $|s| = 1$, then our worker is what we call perfect - though may be malicious. Again, the individual *worker skill*s s_i^* form the vector s_*. With this, the one-coin model assumption can be written as

$$\begin{aligned} \mathbb{P}\left(Y_{ij} = y_j^*\right) &= \frac{1 + s_i^*}{2}, && \text{for } i \in [w], j \in [t] \text{ and} \\ (Y_{uv})_{(u,v)\in A} &\perp (Y_{xy})_{(x,y)\in B} && \text{for } A, B \subset [w] \times [t], A \cap B = \emptyset\,. \end{aligned} \tag{3.1}$$

Finally, it will be convenient to work with the random variable T_{ij} denoting whether the reported worker label for a given task is correct. Define

$$T_{ij} \doteq 2\,\mathbb{I}\left\{Y_{ij} = y_j^*\right\} - 1 \in \{\pm 1\}.$$

Convenient alternative ways to write this are

$$T_{ij} = y_j^* Y_{ij}, \quad Y_{ij} = y_j^* T_{ij}.$$

Note that, $T_{ij} \sim 2\,\mathrm{Ber}\left(\frac{1+s_i^*}{2}\right) - 1$, and therefore $\mathbb{E}\left[T_{ij}\right] = s_i^*$.

The goal. Our main focus is the inference problem: upon observing the labels provided by the workers we wish to find y_*.

More precisely, we wish to find an estimation "procedure", $A : \{\pm 1\}^{w\times t} \to \{\pm 1\}^t$ that achieves low loss, measured as the probability of mislabelling a uniformly randomly chosen task J:

$$\ell(A) \doteq \mathbb{P}\left(A(Y)_J \neq y_J^*\right).$$

The loss of a procedure A will depend on the input distribution; which is fully characterized by the skills s_* of the workers and the true labels y_* of the tasks. Naturally, we wish to find a procedure that has low loss for all possible input distributions. To enable this discussion, we make explicit this dependence: denote by $\ell(A, (s_*, y_*))$ the expected loss of procedure A when run on data conforming to the Dawid-Skene model with parameters (s_*, y_*). We, not unreasonably, ask that a procedure A achieve small loss over any true label setting, that is, we measure its performance as

$$\sup_{y_*\in\{\pm 1\}^t} \ell(A, (s_*, y_*))\,. \tag{3.2}$$

On the other hand, it is also reasonable to expect that procedures are able to perform better when the workers are generally more qualified. In order to capture this, we may wish to compare the performance of A to so-called oracles that are aware of the skill of each worker. The loss associated to the best oracle is captured by

$$\ell^*(s_*) \doteq \inf_A \sup_{y^* \in \{pm1\}^t} \ell(A, (s_*, y_*)), \tag{3.3}$$

where the infimum is over the procedures. Indeed, since s_* is fixed, the best choice for A in this expression can depend on s_*. We will see in Section 3.4 the optimal oracle rule and a tight characterization of the loss it suffers. Observe that comparing $\inf_A \sup_{y_* \in \{\pm 1\}^t} \ell(A, (s_*, y_*))$ to $\ell^*(s_*)$ reveals the additional difficulty of the inference problem introduced by the unknown skills.

Additional notes. There is an inherent ambiguity in this model. Consider the true generating parameters (s_*, y_*) and their negation $(-s_*, -y_*)$, meaning the workers have the same absolute skill but report the opposite answer, and the generating true label also flipped. These are impossible to distinguish based on the observed worker labels. To break this symmetry we assume

$$\sum_i s_i^* > 0, \tag{3.4}$$

that is, on average, the workers are biased towards the correct answer. For the purposes of the analysis we will evaluate our procedures against both the true and flipped labels, and take the better, that is,

$$\ell(A) = \min\{\mathbb{P}\left(A(Y)_J \neq y_J^*\right), \mathbb{P}\left(-A(Y)_J \neq y_J^*\right)\}. \tag{3.5}$$

Eventually we will investigate under what conditions, in addition to Equation (3.4), we are able to differentiate between the two otherwise symmetric label assignments. The final note to make is that for the purpose of the analysis and without loss of generality, we assume that $y_* = \mathbb{1}$. This indeed can be done as long as the estimation method A is not biased towards any label.

3.2 Discussion of the Model

As any result stated is only guaranteed to hold when the data matches our model, it is important to evaluate how realistic the model is. On the other hand, one should not

mistaken the conditions as to being necessary for algorithms to work. In fact, it is very likely that the algorithm can produce good results under weaker conditions, too.

With this in mind, let us discuss the constraints imposed by the model. The independent source (worker) assumption may be violated for a number of reasons. Clearly, if workers copy or simply collaborate they cease to be independent. However, for microtasking problems where each individual instance takes only seconds, with a platform such as Amazon Turk where it is not possible to transfer answers to a large number of instances, this is not likely to be an issue. We also suppose that the performance of the worker is uniform over all instances. This is quite possibly not true, even when the instances are homogeneous in nature. However, experimental work suggests that despite this, such models still lead to significantly better results than the baseline majority voting.

Altogether the model imposes a rigid structure on the labelling process, and, as shown by Gao and Zhou (2013), can lead to performance worse than uniform majority in misspecified models: consider two types of tasks and two groups of workers. The first group labels very well on task type I, and knows nothing of type II, and vice versa. If the two worker groups are equal in size, the number of the instances belonging to each task is imbalanced in a quadratic fashion, and the number of tasks is significantly larger than the number of workers, assuming the Dawid-Skene model and the maximum likelihood estimates suffers loss (measured in expected average number of mistakes made) that converges to zero at a rate only polynomial in the number of instances. In contrast, the error rate of majority voting still converges exponentially fast in the number of workers. Note that while these are stated in terms of different parameters, the above statement about the convergence rate in the Dawid-Skene model implies that it is also polynomial in the number of workers. While an important illustration, the setup used is not typical of simple, homogeneous labelling problems – for these the one-coin model, or more generally the Dawid-Skene model is reasonable, and this is supported by the large number of experimental work showing the benefits of these models in practice.

One final technical assumption made is that every worker labels every instance. This can be relaxed; for instance Zhang et al. (2014) introduce a "labelling probability" specific to each worker in addition to their skills. This is a good model when the worker does not choose to skip instances, and the order in which she was presented with them was chosen uniformly randomly from all orders.

3.3 Performance Baselines

Before presenting results from the literature, we prepare by specifying a number of ways in which they can be evaluated. The first method to consider for the aggregation of the worker labels is (Uniform) Majority Voting (MV): the algorithm that outputs the label that was reported the most often, individually on each task. Note that MV is indifferent to the skills of the workers, therefore one hopes that inference methods that do take skills into account will outperform it. An upper bound on the loss of MV is presented in Section 3.3.1. Afterwards, in Section 3.3.2, we define various stylized types of crowds, or worker collections, on which we evaluate the error bounds associated to the inference methods in the literature. These error rates will be compiled into a table, namely 3.1, at the end of the chapter.

3.3.1 Majority Voting

Uniform Majority Voting is a special instance of the weighted majority voting (WMV) algorithm. We define, for $v \in \mathbb{R}^w$, the v-weighted majority rule to be

$$\begin{aligned} f_v^{\mathrm{maj}} : \{\pm 1\}^w &\to \{\pm 1\} \\ x &\mapsto \operatorname{sgn}\left(x^\top v\right), \end{aligned} \tag{3.6}$$

where x serves as the collection of worker labels on an instance. Note that the scale of the weight vector has no bearing on the decision rule, that is $f_v^{\mathrm{maj}} = f_{rv}^{\mathrm{maj}}$ for all $r > 0$. While the case that $x^\top v = 0$ is not handled by the definition, we will turn a blind eye, and for the purposes of the analysis we will assume this results in an error. We can extend f_v^{maj} to allow weights that have components infinite in magnitude, but only if all such components have the same sign. Finally, by abusing notation, we shall also write f_v^{maj} for the extension of this map to a $\{\pm 1\}^{w\times t} \to \{\pm 1\}^t$ map, where $f_v^{\mathrm{maj}}(Y)_j \doteq f_v^{\mathrm{maj}}(Y_{:j})$, $1 \le j \le t$ and $Y_{:j}$ stands for the vector $(Y_{ij})_{1\le i\le w}$, i.e., the jth column of Y.

Uniform Majority Voting (or simply Majority Voting) is WMV with uniform weights – for instance $v = \mathbb{1}$, – and will be denoted by f^{maj}. We see that this indeed reduces to reporting the label that occurred more often. In Section 5.2.1 (Corollary 2) we show that for a task with true label y_*,

$$\mathbb{P}\left(f^{\mathrm{maj}}(Y) \neq y_*\right) \le \exp\left(-\frac{1}{4}\Big(\sum_{i=1}^{w} \frac{s_i^*}{v_i^*}\Big)^{-1}\Big(\sum_{i=1}^{w} s_i^*\Big)^2\right) \tag{3.7}$$

holds, where v_i^* denotes the log-odds of worker i labelling correctly, that is,

$$v_i^* = \log \frac{1 + s_i^*}{1 - s_i^*}. \tag{3.8}$$

We will later see that the star notation, suggesting –by our convention– that these are "true" weights is in a sense justified. The expression in the exponent of the bound is somewhat convoluted. We offer two simplifications. For the degenerate case of every worker having the same skill $s_i^* = s_0^*$, the exponent becomes $-\frac{1}{4}s_0^* v_0^*$, where $v_0^* = \log \frac{1+s_0^*}{1-s_0^*}$. We will see the same (up to a constant) in Section 3.4.1, where it characterises the error of the optimal decision rule. The coincidence is only under this uniformity assumption. Alternatively, we note $s_i^*/v_i^* = s_i^*/\log \frac{1+s_i^*}{1-s_i^*} \leq 1/2$ no matter the value of $s_i^* \in [-1, 1]$, which leads to

$$\mathbb{P}\left(f^{\text{maj}}(Y) \neq y_*\right) \leq \exp\left(-\frac{1}{2}w\Big(\frac{1}{w}\sum_{i=1}^{w} s_i^*\Big)^2\right). \tag{3.9}$$

This is really loose; we have $s_i^*/v_i^* \to 0$ as $|s_i^*| \to 1$, yet we upper bound it by $1/2$. On the other hand, to the best of our knowledge, this was the bound known for the error of majority voting before; for example Li et al. (2013), in their Corollary 6, offer the same.

3.3.2 Stylized Crowds

We define some special, parametrized collections of workers that will help reveal the behaviour of the bounds proved on the various inference methods. The first two are well-known in the literature, and are commonly used for lower bound arguments; the third is our addition as far as we know. There are three measures of crowds that commonly appear in bounds: the average of the skills ($\bar{s}$), the average of the absolute values of the skills ($\bar{\mu}$), and the average of the squares of the skills ($\bar{\nu}$). We show each of these for every crowd considered.

Uniform. Perhaps the simplest crowd: let each of the w workers have the same skill s. A crowd with this property will be referred to as $\mathcal{C}_s^{\text{U}}$. The associated measures are $\bar{s} = s$, $\bar{\mu} = |s|$, and $\bar{\nu} = s^2$.

Spammer-hammer. Everyone in our crowd either has $s = 0$, a spammer, or $s = 1$, called a hammer. Such crowds demand that inference algorithms be able to differentiate between workers of vastly different skills. This class, $\mathcal{C}_q^{\text{SH}}$, is parametrized by q, the portion of workers that are hammers. We also see that $\bar{s} = q$, $\bar{\mu} = q$, and $\bar{\nu} = q$.

Evil Twins. Out of $2w + 2$ workers w has skill 1, w has skill -1, and 2 have skills $\varepsilon > 0$. This crowd is very knowledgeable, therefore we can expect near perfect labels from inference methods able to differentiate between malicious and honest workers. We use $\mathcal{C}_\varepsilon^{\mathrm{ET}}$ to refer to this type of crowd, parametrized by the bias of the near-spammer workers. In this case $\bar{s} = 2\varepsilon$, $\bar{\mu} = 1 - \frac{2(1-\varepsilon)}{w}$, and $\bar{\nu} = 1 - \frac{2(1-\varepsilon^2)}{w}$.

3.4 Existing Results

We collect, under a unified notation, and discuss the existing theoretical knowledge pertaining to the inference problem in the Dawid-Skene and the one-coin model. This work spans the past years, recently culminating in results claiming (near)-optimality. It has been long known (Nitzan and Paroush, 1982) that weighted majority voting with weights corresponding to the log-odds of the worker labelling correctly leads to optimal decisions. That is, the optimal decision rule f^{OPT} is given by $f^{\mathrm{maj}}_{v_*}$:

$$\operatorname{sgn}\left(x^\top v_*\right) = \operatorname{sgn}\left(\sum_{i\in[w]} x_i \log\frac{1+s_i^*}{1-s_i^*}\right), \tag{3.10}$$

for $x = (x_i)_i \in \{\pm 1\}^w$, $s_* \in [-1,1]^w$. The rule is optimal in the sense that it has loss $\ell^*(s_*)$ (cf. Equation (3.3)); this can be derived from the Neyman-Pearson lemma. However, until recently it was not clear what error this method suffers. Berend and Kontorovich (2014) show an improved upper bound on the probability of mislabelling, as well as an asymptotically matching lower bound. Details of these and additional results can be found in Section 3.4.1. Li et al. (2013) explored the error rate for all hyperplane rules, considering missing worker labels as well. When skills are unknown the problem becomes considerably harder.

Karger et al. (2011b,a), in a sequence of works, proposes various methods to estimate binary labels in the one-coin model, finally achieving minimax optimality in some measure; although, they require a special structure on the allocation of instances to workers. They first notice that the worker response matrix has low-rank with the first left and right singular vectors corresponding to true labels and skills, respectively. Later they replace this estimation method with an almost identical message-passing algorithm for which they can show much stronger, exponential bounds on the error rate. This analysis relies on having a very large number of tasks and workers, however. Details are deferred to Section 3.4.2.

Ghosh et al. (2011), also working in the one-coin model but with (nearly-)full worker response matrices, applies singular value decomposition to $YY^\top$ to obtain bounds similar

to that of Karger et al. (2011b). Dalvi et al. (2013) analyse any allocation in terms of the so-called expansion gap of the bipartite graph on instances and workers induced by the labelling. They show finite sample bounds on the error of the skill estimates measured in squared ℓ_2 norm, which in the special task allocation of Karger et al. (2011b) leads to an exponential bound on the expected number of mistakes. Jaffe et al. (2015), working in the two-coin model, use spectral methods on the same matrix as Ghosh et al. (2011) do and show asymptotic rates on the recovery of the confusion matrix.

Another approach taken by Gao and Zhou (2013); Zhang et al. (2014) is the analysis of a regularized log-likelihood function associated to this model, and the EM algorithm that attempts to find its maximizing parameters: the generating skills and true labels. The log-likelihood function corresponding to this model is not convex, therefore EM is not guaranteed to find maximum likelihood (ML) estimates. Moreover, the ML estimate of a model does not automatically enjoy low loss. In a paper only published on arXiv (though cited a number of times) Gao and Zhou (2013) claims a minimax optimal exponential bound on the loss for both the global maximizer of the regularized log-likelihood function and the estimate returned by a slightly modified version of EM. We have found a mistake in the proof of the former and communicated it to the authors – who have confirmed it, while the argument for the latter bound stands. In fact, Zhang et al. (2014) use the same analysis of the same modified EM algorithm, now in the multiclass and fully general Dawid-Skene setting, to claim near-optimal sufficient conditions for a perfect label reconstruction. For EM to be successful with high probability, a special, spectral-based initialization is used. Details appear in Section 3.4.3. The bound we prove in Chapter 5 is on the global maximizing parameters of the likelihood function for the one-coin model. As such, we fill the gap that opened in (Gao and Zhou, 2013); though, our bound is presented in a different form.

Chapter 4 introduces a number of topics, including the log-likelihood function, EM and concentration inequalities familiarity with which may be needed for the more in-depth discussions in the following sections.

3.4.1 Known or Approximate Parameters

We start with studying the simpler case when the skills are known, or we are given estimates of them, and only the true labels remain to be inferred. The resulting rates serve as a benchmark, and also reveal quantities that characterize the effectiveness of a crowd of workers. The definitive work on this topic is that of Berend and Kontorovich (2014). They work in the one-coin model, and are concerned with the error made by the optimal decision

rule, $f^{\mathrm{OPT}} : \{\pm 1\}^t \to \{\pm 1\}$, given by Equation (3.10). It is shown that the so called *(total) committee potential*

$$\Phi = \sum_{i \in [w]} s_i^* \log \frac{1 + s_i^*}{1 - s_i^*} \tag{3.11}$$

governs the probability of making an error. For a single task (i.e., $t = 1$) we have

$$\mathbb{P}\left(f^{\mathrm{OPT}}(Y) \neq y_*\right) \leq \exp\Big(-\frac{1}{2}\,\Phi\Big), \tag{3.12}$$

$$\mathbb{P}\left(f^{\mathrm{OPT}}(Y) \neq y_*\right) \geq \frac{3}{4[1 + \exp(2\,\Phi + 4\,\sqrt{\Phi})]}\,. \tag{3.13}$$

In short, we see that

$$-\log \mathbb{P}\left(f^{\mathrm{OPT}}(Y) \neq y_*\right) \asymp \Phi\,,$$

where $\asymp$ denotes equivalence up to universal multiplicative constants. Both the lower and upper bounds are contributions of this paper. The upper bound is proven using Kearns and Saul's inequality (Lemma 3), which is tighter in this setting than other commonly used concentration bounds, and is therefore of interest to us as well. Note that the committee potential increases with each additional (non-spammer) worker; sometimes we may wish to talk about the *average committee potential*, $\frac{1}{w}\sum_{i \in [w]} s_i^* v_i^*$, denoted by $\overline{\Phi}$, instead. This notion is especially informative when analysing the effects of the quality of the crowd and the number of workers separately.

In addition to the known-skill setting, the scenario where the skills are estimated on a number of golden units, instances with known labels, is also studied, both from a frequentist and a Bayesian point of view. In the frequentist approach, one option is to replace the true skills in the weighted majority decision rule by the estimates $\hat{s} \in [-1, 1]^w$, yielding weights $\hat{v}_i = \log \frac{1+\hat{s}_i}{1-\hat{s}_i}$ and the rule $f_{\hat{v}}^{\mathrm{maj}}$. We base the proof of our Lemma 11 on the argument of Berend and Kontorovich who showed a finite sample bound on the mislabelling probability of $f_{\hat{v}}^{\mathrm{maj}}$. The form of their bound is as follows. For any $0 < \delta < 1$, $0 < \varepsilon < \min\{5, 2\overline{\Phi}\}$, letting t_i denote the number of golden units worker i was tested on, if

$$t_i(1 - |s_i|) \geq 6 \left(\frac{\sqrt{4\varepsilon + 1} - 1}{4}\right)^{-2} \log \frac{4w}{\delta}\,,$$

then

$$\mathbb{P}\left(f_{\hat{v}}^{\mathrm{maj}}(Y) \neq y_*\right) \leq \delta + \exp\left[-w \frac{(2\overline{\Phi} - \varepsilon)^2}{8\overline{\Phi}}\right].$$

Here δ is associated to the probability of receiving a non-representative sample of worker labels on the golden units, ε captures how much we err when estimating weights. Note that

the bound captures the additional error suffered over that of the optimal oracle decision rule due to having to estimate the skills.

3.4.2 Inference with Task Allocation

In this section we discuss inference results that require a special worker-instance, or worker-task pairing. This allocation of tasks to workers, however, is designed statically before any labelling takes place, the workers then provide their labels on the tasks that were assigned to them, and finally an algorithm aggregates these into estimates of the true labels. Karger et al. (2011b,a, 2013, 2014) design inference methods in this setting. The results that they derive have since been superseded by other works, which do not even require such specific assumptions on the structure of labelling. Nonetheless, it is educational to examine the methods they propose as well, for they provide a different perspective on the same problem. Moreover, while not our current focus, these works also allude to adaptive (dynamic) task allocation. Karger et al. (2011b) characterize their crowd by the quantity $\bar{\nu} = \frac{1}{w}\sum_{i\in[w]}(s_i^*)^2 = \frac{1}{w}\|s_*\|_2^2$, and show upper and lower bounds (for relatively weak crowds) on the mislabelling probability of their methods in the one-coin model, with $\bar{\nu}$ appearing in the exponent of both. Another important quantity they work with is the number of times each task is labelled, l, in their task-allocation. This is considered to be small; in fact their final results are phrased in the form of bounds on l when other problem parameters, including the probability of mislabelling, are fixed. The first paper (Karger et al., 2011b) introduces the instance allocation procedure, and uses singular value decomposition on the collected worker labels to recover the skills and true labels. The resulting upper bound on the loss ℓ (cf. Equation (3.2)) is only in the order of $\frac{1}{l\bar{\nu}}$. The following paper (Karger et al., 2011a) replaces the inference method by a message passing algorithm that is almost identical, but allows for an analysis leading to a bound on the loss that is now on the scale of $\exp(-l\bar{\nu})$. A significantly extended version of the same conference paper was published some time later (Karger et al., 2014). Finally, Karger et al. (2013) reduce the multi-class, general Dawid-Skene problem to a number of binary tasks, for which they call their already proposed estimation, yielding a near minimax optimal algorithm.

Instance allocation. Intuitively, the instance allocation proposed aims to spread the workers out evenly over the instances. An allocation will be represented by a bipartite graph, with vertices on the left denoting instances, and those on the right workers. There is an edge between an instance and a worker if the worker will provide a label for the instance. As mentioned, each instance is labelled by l workers, and each worker labels r

instances. (Clearly, we require $lt = rw$.) We construct a random, so-called (l, r)-regular bipartite graph on these $t + w$ vertices. Start by attaching "half edges" to each task and worker vertex. Order all half-edges coming from instances in the natural (or any fixed) order, and pick a random order (uniformly over all possible orders) for the half-edges attached to worker vertices. Join the half-edges that sit in identical locations of the two orders. It is known from random graph theory that the resulting graph, especially when its large in size compared to l and r, and therefore is sparse, enjoys some desirable properties. Karger et al. (2011b) use that this graph has large spectral gap, which aids in separating the signal from the noise in the worker response matrix when singular value decomposition is used. Karger et al. (2011a) use that these graphs become locally tree-like for large systems: if we denote by $G_{v,k}$ the set of vertices reachable from vertex v in k steps along the edges of the graph, one can prove that $G_{v,k}$ is a tree with probability no more than $[(l-1)(r-1)]^{k-1}\frac{3lr}{t}$. This becomes important to the analysis of the inference algorithm.

Inference Algorithm. We focus only on the message passing algorithm of Karger et al. (2011a). The method takes as argument a worker label matrix A; n_i and n_j denote the neighbourhoods of the worker vertex i and instance vertex j, respectively. The algorithm iteratively passes "messages" between the partitions of the graph, with each vertex, in a sense, broadcasting its "belief" on the true value of the parameter it represents. We identify two types of messages, corresponding to where they originate from. Task messages, $x_{\{j\to i\}}$, represent the log-likelihood of task j having a positive label. Worker messages, $y_{\{i\to j\}}$, represent how reliable worker i is. To calculate a new message $\{a \to b\}'$, we consider all incoming messages to a, except the one from b, and sum those up with the signs set by the appropriate entry from A. This leads to the algorithm:

1. Initialize worker messages as $y_{\{i\to j\}} \sim \mathcal{N}(1, 1)$, the normal distribution with mean and variance 1.
2. In each iteration, update the messages. For every (i, j) edge:
   (a) $x'_{\{j\to i\}} \leftarrow \sum_{i'\in n_j\setminus\{i\}} A_{i'j}y_{\{i'\to j\}}$
   (b) $y'_{\{i\to j\}} \leftarrow \sum_{j'\in n_i\setminus\{j\}} A_{ij'}x_{\{j'\to i\}}$
3. Output the estimates from weighted majority voting:

$$\operatorname{sgn}\left(\sum_{i\in n_j} A_{ij}y_{\{i\to j\}}\right).$$

Regarding the number of iterations, k: there are multiple bounds offered in the paper, one is in terms of k; another requires that k reaches a threshold – specified in terms of other problem parameters, – above which the convergence rate displayed becomes independent of the iteration number.

Karger et al. (2014) draw the attention of the reader to the fact that if the omitted term in the sums of the update rule were included, this algorithm would calculate the exact same quantity as that of Karger et al. (2011b). In practice, the two algorithms behave similarly. However, in terms of the analysis this difference seems crucial.

Proof Technique. Here we describe the proof technique used by Karger et al. (2014). Fix (or, equivalently, uniformly randomly select) a task vertex j, and denote the random variable corresponding to its label estimate after k iterations by $\hat{X}^{(k)}$. Recall, the loss is defined as $\mathbb{P}\left(\hat{X}^{(k)} \neq (y_*)_j\right)$. For the purposes of the analysis, we automatically make an error if (1) at vertex j, the graph is *not* locally tree-like up to $2k-1$ steps, (2) $\hat{X}^{(k)}$ is incorrect, given that the graph is locally tree-like. The result mentioned regarding the locally tree-like structure of these bipartite graphs is used to bound the probability of the first event. This term will vanish when the number of tasks is large, leaving the second term to be dominant. A technique called density evolution is used to bound the probability of the second event; the intuition behind it is not complicated. Note that in k iterations of the inference algorithm information from only the closest $2k-1$ vertices can propagate to j; all other vertices, therefore, have no influence on the label reported for that instance. This justifies focusing only on this local neighbourhood of j. The requirement that the local neighbourhood is a tree guarantees the independence of the random variables corresponding to the messages.

We offer additional intuition; this is along the lines of the justification used for the so-called belief propagation algorithms often used for inference. Due to the design of the update rules, the spread of "information" in the tree itself exhibits structured behaviour. For ease of communication let us root the tree at j; the reader may wish to follow the argument on Figure 3.2. We will refer to vertices by depth: the root, vertex j, being at depth 0 and the last layer on the figure at depth 3. Recall that the label estimate $\hat{X}^{(k)}$ was the sign of the weighted average of the incoming worker messages, $\operatorname{sgn}\left(\sum_{i\in n_j} A_{ij} y_{\{i\to j\}}\right)$. Visually, these messages are directed from depth 1 to 0. Consider one of the workers, say i, sending such a message $y_{\{i\to j\}}$. In the last iteration of the algorithm, this was calculated as $\sum_{j'\in n_i\setminus\{j\}} A_{ij'} x_{\{j'\to i\}}$. Notice that it only aggregates information from its children (deeper in the tree) due to the exclusion of the $x_{\{j\to i\}}$ message. In turn, applying this same

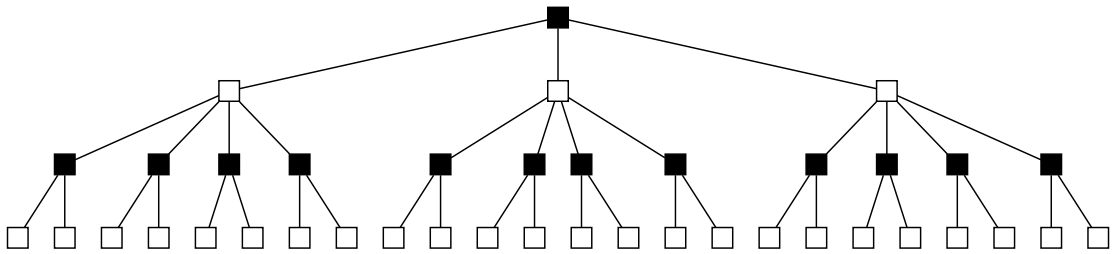

Figure 3.2: The locally tree-like structure of Karger et al.'s task allocation, for $k = 1$.
For this representation we "folded out" the bipartite graph to reveal the tree. Black vertices represent instances, white ones workers. Note, we set $l = 3$ and $r = 5$ for this illustration.

argument to these incoming messages, we see that $y_{\{i\to j\}}$ only aggregates information from the subtree rooted at it. Continuing with this argument, we conclude that this message passing algorithm is equivalent to propagating the messages upwards from the leaves, layer by layer, finally leading to a label estimate at the root.

In the technical analysis the distribution of the messages is studied from iteration to iteration. With each one, the signal -the assumed bias towards the correct label- gets amplified, and the message distributions concentrate around their true values, distinguishing good and bad workers, positive and negative labels with ever increasing confidence.

We now quantify these statements, quoting Theorem 1 and Corollary 1 of Karger et al. (2014). First, for brevity, let $\hat{l} = l - 1$ and $\hat{r} = r - 1$, and define

$$\sigma_k^2 \doteq \frac{2\bar{\nu}}{\bar{\mu}(\bar{\nu}^2\hat{l}\hat{r})^{k-1}} + \left(3 + \frac{1}{\bar{\nu}\hat{r}}\right)\frac{1-(1/\bar{\nu}^2\,\hat{l}\hat{r})^{k-1}}{1-(1/\bar{\nu}^2\,\hat{l}\hat{r})}\,,$$

which actually controls the variance of the sub-gaussian tail of the estimate $\hat{X}^{(k)}$.

Theorem 1. *If $\bar{\mu} > 0$ and $\bar{\nu}^2 > 1/(\hat{l}\hat{r})$, then for any $y_* \in \{\pm 1\}^t$, the estimate after k iterations of the algorithm $\hat{y}^{(k)}$ achieves, on the uniformly randomly chosen task J,*

$$\mathbb{P}\left(\hat{y}_J^{(k)} \neq y_J^*\right) \leq e^{-l\bar{\nu}/(2\sigma_k^2)} + \frac{3lr}{t}(\hat{l}\hat{r})^{2k-2}.$$

For a simpler presentation of the result they make a number of observations. First, the second term vanishes with large t. Second, under the assumption $\bar{\nu}^2\hat{l}\hat{r} > 1$, for large k, σ_k^2

converges linearly to the finite

$$\sigma_\infty^2 = \Big(3 + \frac{1}{\bar{\nu}\hat{r}}\Big)\frac{\bar{\nu}^2\hat{l}\hat{r}}{\bar{\nu}^2\hat{l}\hat{r} - 1}\,.$$

This leads to the following corollary of the above theorem:

Corollary 1. *Under the same hypothesis, for* $t_0 = 3lre^{l\bar{\nu}/(4\sigma_\infty^2)}(\hat{l}\hat{r})^{2(k-1)}$ *and* $k_0 = 1 + \Big(\log(\bar{\nu}/\bar{\mu}^2)/\log(\hat{l}\hat{r}\bar{\nu}^2)\Big)$ *we have*

$$\mathbb{P}\left(\hat{y}_J^{(k)} \neq y_J^*\right) \leq 2\exp\left(-\frac{l\bar{\nu}}{4\sigma_\infty^2}\right) \tag{3.14}$$

for all $t \geq t_0$ *and* $k \geq k_0$. *Observe that in* $l\bar{\nu}/(4\sigma_\infty^2)$ *the term* $4\sigma_\infty^2$ *becomes negligible asymptotically as* $t \to \infty$.

In the original paper the brackets are missing around $4\sigma_\infty^2$ in the expression for t_0; we added them here. The authors note that a shortcoming of this bound is that t_0 is impractically large. They point out, though, that in practice the inference method seems to exhibit similar performance on smaller problem sets as well.

Towards better understanding the problem, the authors prove minimax lower bounds on the loss: we are interested in how much error the best inference method makes in a worst-case sense. In the context of task allocation, we also allow to optimize over assignment schemes, though we parametrize them by l, the number of times the most labelled instance is labelled. The set $\mathcal{T}_l$ collects all assignment schemes with this property. Additionally, the loss suffered by any inference method depends on the crowd of the workers providing the labels; therefore we evaluate the estimation procedures over crowd classes characterized by some measure of the crowd quality. Motivated by the appearance of $\bar{\nu}$ in the upper bound, the minimax rate will be parametrized by the same. Similarly, the loss has to be measured over all possible generating true labels. Precisely stated, the minimax loss studied is

$$\min_{\tau\in\mathcal{T}_l, A}\ \max_{y_*, s_*\in\mathcal{P}_q}\ \ell(A)\,,$$

where A ranges over all possible inference methods, ℓ depends on τ, y_*, s_* (suppressed in the notation) and $\mathcal{P}_q = \{s_* \mid \bar{\nu}(s_*) = q\}$ denotes worker collections with $\bar{\nu}$ equal to a prescribed q. Using the spammer-hammer crowd $\mathcal{C}_q^{\mathrm{SH}}$ and the estimator that is aware of all skills, the authors get a lower bound on the minimax rate:

$$\min_{\tau\in\mathcal{T}_l, A}\ \max_{y_*, s_*\in\mathcal{P}_q}\ \ell(A) \geq \frac{1}{2}\exp\left(-(q+q^2)l\right),$$

shown in the $q \leq 2/3$ domain; we will see later that we in fact require another measure for crowds with high quality, and that this inference method is not order-optimal in such settings. Towards establishing the minimax optimality of their method in the given domain we observe that $q^2 \leq q \leq 1$, therefore the exponent asymptotically matches $-ql = -\bar{\nu}l$. We do note that one may consider other parametrizations of crowd collections that the max is taken over, that are perhaps sensitive to other quality measures of the crowd.

Our additional contribution to understanding the performance of this method is materialized in the evaluation of the error bound on our collection of typical crowds. For a detailed comparison between inference methods see Section 3.4.4. The fact that the uniform majority voting (MV) algorithm has tighter bounds than this method is understandable; MV works perfectly on such crowds. But notice that on spammer-hammer crowds the MV may still outperform the inference method of this paper. Namely, if $1/q \leq 4\frac{r}{r-1}$ holds, the upper bound on the loss of majority voting is tighter than that on this message passing algorithm. (Derived from setting the exponent of the bound on MV to be larger than that of this algorithm.) This may come as a surprise, especially considering that Karger et al. (2014) conclude, as a consequence of a lower bound on the loss of MV (Lemma 2), that majority voting performs worse than their message passing algorithm. Indeed, in many scenarios that is the case, but not all, and while this may be captured by their exact result the discussion seems to omit the necessary qualifications. It may be interesting to investigate why the error bound associated to this message passing algorithm suggests poor performance in some scenarios with high quality workers; to understand whether it is a failure of the inference method, or the analysis. The authors do note that they found practical datasets often have $\bar{\nu} \approx 0.3$, but that already renders the above condition on majority voting outperforming their algorithm true. (Recall, for the spammer-hammer crowd $\mathcal{C}_q^{\mathrm{SH}}$ we have $\bar{\nu} = q$.) Nonetheless, we do exhibit a crowd of high quality, $\mathcal{C}_\varepsilon^{\mathrm{ET}}$, on which the message passing algorithm does perform significantly better than majority voting.

Finally, let us mention that Liu et al. (2012) observed that their message passing algorithm can be realized as belief propagation (BP) with a specific prior over the skills. Belief propagation is a heuristic algorithm for finding the maximum a posteriori probability (MAP) estimate. This estimate is closely related to the maximum likelihood (ML) estimate, establishing an interesting connection between their work and the methods we study in the next section.

3.4.3 General Inference

Two closely related works (Gao and Zhou, 2013; Zhang et al., 2014) provide bounds on the loss suffered by inference methods working directly with maximum likelihood estimates.

Gao and Zhou (2013) consider the objective function that EM optimizes, which may be interpreted as a regularized log-likelihood function, and claim exponential bounds, stronger than any other appearing in the literature, on the loss of its global optimizer. They also show that despite this objective function not being convex, under some mild conditions a well-initialized EM finds label estimates that enjoy practically the same statistical guarantees as the global optimum. We show that their proof regarding the global optimizer is incorrect; the bound on the estimates returned by EM appears to hold.

In order to study the global optima of the objective function F that EM optimizes, Gao and Zhou (2013) study the expected behaviour of F, and find its maximizing parameters $\tilde{\theta} = (\tilde{s}, \tilde{y})$. The proof first provides guarantees on the mislabelling probability when $\tilde{y}$ is used, then establishes that this error, with high probability, controls that of the maximizing parameters of the random F we observe. We find a mistake in a crucial step of this latter argument. We showcase a counterexample to one of the steps, included in Appendix B, and in personal communication with the authors establish that the (sub)-result, Lemma F.1, whose proof contains this step "is not correct. As [we] have pointed out, the left-hand side of the inequality grows in a polynomial rate, while the right-hand side grows only logarithmically." (priv. comm. with Gao and Zhou (2013)).

The proof that Gao and Zhou (2013) provide guaranteeing the performance of the actual estimates returned by the EM algorithm is independent of this result. It relies on two main steps; (1) they show that if given a "good enough" initializer, EM in just one iteration will find label estimates that enjoy the strong exponential mistake bounds. Then they (2) propose an initialisation method that achieves the required accuracy for step (1) to be applied. We note that accepting this result as true "very likely" implies that the global optimizer enjoys at least the same bounds, if not better. To make this rigorous one may show that the global optimizer is a good enough initializer, then vacuously applying step (1) (applying an EM iteration does not change the parameters we are considering as we are already at a local maximum) yields the exponential bounds. Zhang et al. (2014) apply the same proof technique to show, among other things, sufficient conditions on the number of tasks and workers needed in order to recovering all labels perfectly in the general Dawid-Skene model by using EM with a spectral based initialisation.

3.4.4 Comparison of Bounds

We collect in this section the bounds claimed on the probability of mislabelling by highlighted works, each evaluated on our collection of stylized crowds (cf. Section 3.3.2). This information, displayed in Table 3.1, highlights for what domains the various proposed

Method	$\mathcal{C}_s^{\mathrm{U}}$	$\mathcal{C}_q^{\mathrm{SH}}$	$\mathcal{C}_\varepsilon^{\mathrm{ET}}$
MV	$ws \log \frac{1+s}{1-s}$	$\frac{wq}{1/q-1}$	$\frac{\varepsilon^2}{1-2\varepsilon}$
MP	$\frac{ws^2}{3+1/(ws^2)}$	$\frac{wq}{3+1/(wq)}$	$\frac{w\alpha_2}{3+1/(w\alpha_2)}$
EM-Spec	$ws \log \frac{1+s}{1-s}$	$w \max\left(q, \frac{1}{3} q \log \frac{1+q}{1-q}\right)$	$w \max\left(\alpha_2, \frac{1}{3}\alpha_1 \log \frac{1+\alpha_1}{1-\alpha_1}\right)$
Oracle	$ws \log \frac{1+s}{1-s}$	∞	∞

Table 3.1: Comparison of the guarantees provided for various algorithms in the literature. We consider a number of algorithms with theoretical guarantees on the probability of mislabelling; the algorithms and the bounds for them are precisely stated in Section 3.4.4. We evaluate each of these methods on the typical crowds introduced earlier, displaying the negative exponent of the resulting guarantees in this table. We use $\alpha_1 \doteq 1 - \frac{2(1-\varepsilon)}{w}$, $\alpha_2 \doteq 1 - \frac{2(1-\varepsilon^2)}{w}$; both quantities go to 1 with w increasing.

methods fit well. Here we list the algorithms, as well as the negative exponent, up to a constant, of the best known upper bound on the mislabelling probability of each:

- MV: the (uniform) majority voting algorithm f^{maj} introduced in Section 3.3.1. The negative exponent of the best known upper bound is $\left(\sum_{i=1}^w \frac{s_i^*}{v_i^*}\right)^{-1}\left(\sum_{i=1}^w s_i^*\right)^2$, as given in Equation (3.7).

- MP: the message passing algorithm of Karger et al. (2011b). While the negative exponent is often quoted as $l\bar{\nu}$, it is more specifically

$$l\bar{\nu}\left((3 + \tfrac{1}{(r-1)\bar{\nu}})\tfrac{(r-1)(l-1)\bar{\nu}^2}{(r-1)(l-1)\bar{\nu}^2-1}\right)^{-1},$$

  as seen in Equation (3.14). The variables l and r denote the number of times each task is labelled, and the number of times each worker labels, respectively. We tie $l = r - 1$, and upper bound the exponent by $\frac{l\bar{\nu}}{3+1/(l\bar{\nu})}$.

- EM-Spec: the EM algorithm of Gao and Zhou (2013), initialized with the spectral methods. We use $w \max\{\bar{\nu}, 1/3\bar{\mu} \cdot v(\bar{\mu})\}$ as the negative exponent, which just upper bounds the true bound.

- Oracle: The optimal decision rule with access to the true skills. The negative exponent is the total committee potential Φ, as shown by Berend and Kontorovich (2014).

The definitions of $s_i^*, v_i^*, \bar{\nu}, \bar{\mu}$ are as usual. In order to calculate the entries in the table we simply substitute the appropriate values into these bounds. There are two things to be noted: in the bounds of Karger et al. (2011b) it is l, the number of times each task was labelled, that appears, as opposed to the number of workers. To be able to compare the results we use the latter, w, in the table; they are typically just a constant off anyway.

Before comparing the performances of these bounds to each other, we remark that q should not be taken to its extreme value of 1, as the resulting expressions will not necessarily be representative of the true exponents. With this in mind, we may begin evaluating the algorithms. We first observe that in each column the bounds get stronger as we move down the rows, except for the message passing algorithm. Already on uniform crowds it performs significantly worse than other methods, which becomes especially pronounced when the workers are skilful. (Karger et al. (2011b) do specify they are more concerned with the case when this is not true.) Even on spammer-hammer crowds MV may outperform MP, which is surprising as MV is blind to skills. This is not typical, however; over most problems MP would actually enjoy better guarantees. In fact Gao and Zhou (2013) in their Theorem 4.1 show that for a spammer-hammer crowd with $q = \frac{w^\delta}{w}$, $\delta < 1/2$, majority vote becomes inconsistent. We see that MP still recovers the labels when w is taken to infinity. On the Evil-Twins crowd the inability of MV to differentiate between workers of different skills is fully captured by the almost vacuously true upper bound on its performance.

We also highlight that EM-Spec captures exactly what MP missed: for crowds with very skilful workers, $\bar{\mu} \cdot v(\bar{\mu})$ becomes the dominating measure of labelling correctness. Indeed, observing the exponent of EM-Spec in the spammer-hammer setting we see $\max\left(q, \frac{1}{3} q \log \frac{1+q}{1-q}\right)$ is q until about > 0.93, and only afterwards does the second term become larger. Finally, this second term in the bound of EM-Spec is related to the committee potential; the difference is that as opposed to considering the average committee potential, it calculates the average skill and bounds the error with the committee potential corresponding to that average skill. These may be vastly different, the latter providing weaker guarantees.

Chapter 4

Background

In this chapter we discuss some concepts and techniques that are required for our our main result. A seasoned reader may wish to skip this. Context and motivation, as well as some specific results are provided for each topic.

4.1 The Log-Likehood Function

When studying parametric statistical models, it may be of interest to infer the "generating" parameters of our observations. To be precise, consider a statistical model $(S, \mathcal{P})$, where S is the sample space and $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a parametrized collection of probability distributions on S. We consider the problem of observing a sample from one particular but unknown distribution in this collection, and having to identify which one. Sometimes this may be phrased as inferring or estimating the true parameter, though these two wordings may imply different criteria by which we measure success. For example, in the one-coin model the sample space is the set of all possible worker label matrices of the appropriate size, the parameter space is $\Theta = [-1, +1]^w \times \{\pm 1\}^t$ – the collection of all possible skill and true label vectors. P_θ is induced by our assumption that workers label independently on different tasks and of each other, with the probability of doing so correctly governed by their skill. Indeed, our goal is to find the "generating" parameter, in particular the true labels, of the distribution we are observing.

The likelihood function and its maximizer, the maximum likelihood (ML) estimate are standard tools for doing so. For the rest of the discussion assume that the sample space S is finite; this is sufficient for our application and saves us from technicalities. Nonetheless,

analogous constructs and statements work in general. Given this assumption, we may define, for $x \in S$ $p_\theta(x) = P_\theta(\{x\})$, the probability mass function associated to P_θ. The likelihood, in this case, is a function on $\Theta \times S$ that associates to each element x of the sample space and model parameter θ the probability of observing x in the distribution P_θ:

$$\mathcal{L}(\theta \mid x) = p_\theta(x).$$

The "conditional" in the likelihood is purely notational. When applying this technique, say to worker label matrices in the one-coin model, we are presented with a matrix x, and $\mathcal{L}(\theta \mid x)$ is interpreted as a measure of the "likelihood" that the distribution P_θ is its source. The parameters with the largest likelihood, if any exist, are the ML estimates associated to x:

$$\arg\max_\theta \mathcal{L}(\theta \mid x).$$

While this may be a set of parameters, we work with any arbitrary one. Returning to the motivation of this definition, we note that the distributions corresponding to the ML estimates minimize the so called KL divergence of P_θ from the empirical distribution induced by our observation. This quantity can be interpreted as the amount of information lost when approximating the empirical distribution by P_θ; and therefore the ML estimates, in this sense, are the best approximations available in the model to our observations. On the other hand, this does not, in general, impose any constraints on the proximity of the ML estimates and the true generating parameters when measured directly in the parameter space.

To be precise, we identify three variants of the likelihood function. We already defined $\mathcal{L}(\theta \mid x)$; $\mathcal{L}(\theta \mid X)$, also abbreviated to $\mathcal{L}(\theta)$, will denote the associated random function; while $\overline{\mathcal{L}}(\theta)$ will be the expected value of the latter. Terminology-wise, $f_{\mathcal{P}}^{MLE} : S \to \Theta$ by $x \mapsto \arg\max_{\theta\in\Theta} \mathcal{L}(\theta \mid x)$ is referred to as the maximum likelihood estimator (MLE), as it really is an inference method, a map from the sample space to the parameter space. Further, $\hat{\theta} = f_{\mathcal{P}}^{MLE}(X)$, the estimator applied to the sample results in the (random) estimate $\hat{\theta}$. To apply the MLE it is not necessary that X is generated from P_θ for some $\theta \in \Theta$, but often this is assumed for the sake of the analysis. Note that instead of the likelihood function, it is common to work with the (negative) log-likelihood function (LLF), which simply is the (negative) natural logarithm of the likelihood function. Optimizing over either is equivalent, since log is a convex function.

Now we turn our attention to actually finding the ML estimates. As usual, one can find the critical points of a differentiable LLF by setting the gradient of the function to 0; it remains to select the global maxima. When this cannot be done analytically, an alternative option is to do so numerically. The so-called EM algorithm is a popular numerical

approach for the cases when the model can be phrased in the following form: there is some unobserved, latent data Z and observed data X (for our purposes both discrete), with joint probability mass function $p_\theta(X, Z)$ parametrised by $\theta \in \Theta$. We want to find the MLE of the marginal likelihood of the observed data, $\mathcal{L}(\theta \mid X)$, defined as the likelihood of the marginal of the observed data $\sum_z p_\theta(X, z)$. Again, instead of exact calculations, we take the numerical approach of iteratively finding estimates of the marginal likelihood of the observed data and its maximizer. That is, initializing with some parameter θ_0, we find the expected value of the log-likelihood with respect to the conditional distribution of Z given X, under parametrisation θ_0 as a function of θ, i.e., the function $\theta \mapsto \sum_z p_{\theta_0}(z|X) \log p_\theta(X, z)$. Next we maximize this function to get θ_1, which we use in the next iteration. It is shown that this procedure converges to a local maximum of the (marginal) likelihood function. If the likelihood function is known to be convex, for example, then this coincides with a ML estimate. In general, however, there is no way to guarantee finding an actual ML estimate. In practice many different initializations are tried and the best result is selected.

In the one-coin model we cannot directly find the MLE, so we turn to the EM algorithm. We treat the true labels as the latent data and the skills remain parameters. To do so, we need to assume a distribution on the true labels; the uniform distribution is a sensible choice. Now, from an initial guess on the skill vector the corresponding distribution over the true labels can be calculated. Using these we calculate the marginal likelihood of the worker labels we observed, finally maximizing over skills. We restart with this new skill estimate. At any point the intermediate distribution over the true labels serve as soft labels, with its rounded value used as the estimate for the true labels.

4.2 Concentration Inequalities

When studying random variables one elemental question is how much, and with what probability they deviate from a certain value, such as their expectation (if it exists). Concentration inequalities are statements probabilistically bounding such deviations. Perhaps one of the most basic concentration inequalities is Markov's: for any real-valued, non-negative integrable random variable X and $a > 0$ we have

$$\mathbb{P}(X > a) \leq \frac{\mathbb{E}[X]}{a}.$$

In this thesis we will consider the (random) log-likelihood function associated to the one-coin model, and will want to study its deviation from its expected value; concentration inequalities will prove useful to us. We first state two well known inequalities, Hoeffding's

and its generalization McDiarmid's. Then we introduce the so-called, more recent, Kearns and Saul's inequality, which refines Hoeffding's inequality. We rely heavily on this result in our argument. Finally, we note that Berend and Kontorovich (2013) prove a further improved variant of the same inequality.

Lemma 1 (Hoeffding's Inequality)**.** *Let $(X_i)_{i=1}^n$ be independent, almost surely bounded real-valued random variables with*

$$\mathbb{P}\left(X_i \in [a_i, b_i]\right) = 1 \quad \forall i \in [n].$$

Let $\overline{X} \doteq \frac{1}{n}\sum_{i=1}^n X_i$, the empirical mean, then for all $t \geq 0$

$$\mathbb{P}\left(\overline{X} - \mathbb{E}\left[\overline{X}\right] \geq t\right) \leq \exp\left(-\frac{2n^2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

On a high level, Hoeffding's inequality states and quantifies the statement that for most (typical) samples the sample mean is close to the expected mean. Note that, as stated above, it (equivalently) bounds the probability of receiving a so-called "bad sample" on which the empirical mean is larger than the expected, by more than the allowed margin. By applying the inequality to the negations of the X_i we would bound the probability of seeing samples on which the empirical mean is smaller than the expected, by more than the allowed margin. Finally, considering the event that either of these happen, we arrive at another commonly appearing form of the conclusion of Hoeffding's inequality:

$$\mathbb{P}\left(|\overline{X} - \mathbb{E}\left[\overline{X}\right]| \geq t\right) \leq 2\exp\left(-\frac{2n^2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

McDiarmid generalizes Hoeffding's Inequality by noticing that in place of the empirical mean we may use any function that has bounded dependence on each of its individual arguments.

Lemma 2 (McDiarmid's Inequality)**.** *Let $(X_i)_{i=1}^n$ be independent random variables taking values in the set $\mathcal{X}$. If $f : \mathcal{X}^n \to \mathbb{R}$ is such that there exists $(c_i)_{i=1}^n$ with*

$$|f(x_1, \ldots, x_i, \ldots, x_n) - f(x_1, \ldots, x_{i'}, \ldots, x_n)| \leq c_i$$

for all $i \in [n]$ and all $x_1, \ldots, x_n$, $x_{i'} \in \mathcal{X}$, then

$$\mathbb{P}\left(f(X) - \mathbb{E}\left[f(X)\right] \geq t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n c_i^2}\right).$$

These inequalities, however, prove to be weaker than desired in our setting. Instead, we use an inequality and its corresponding concentration bound proposed by Kearns and Saul (1998), originally for Bernoulli random variables. We quote a variant of the concentration bound, from Raginsky and Sason, 2013, Theorem 2.2.6, that is a simple extension of the original to any bounded real-valued random variable.

Lemma 3 (Kearns and Saul's inequality)**.** *For all $p \in [0,1]$ and $t \in \mathbb{R}$,*

$$(1-p)e^{-tp} + pe^{t(1-p)} \leq \exp\left(\frac{1-2p}{4\log\frac{1-p}{p}}t^2\right). \tag{4.1}$$

Lemma 4 (Kearns and Saul's inequality)**.** *Let $(X_i)_{i=1}^n$ be independent, real-valued random variables with $X_i \in [a_i, b_i]$ almost surely for every $i \in [n]$. Let $\overline{X} \doteq \frac{1}{n}\sum_{i=1}^n X_i$. Then, for all $t \geq 0$,*

$$\mathbb{P}\left(\overline{X} - \mathbb{E}\left[\overline{X}\right] \geq t\right) \leq \exp\left(-\frac{t^2 n^2}{4\sum_{i=1}^n c_i (b_i - a_i)^2}\right),$$

with

$$c_i = \begin{cases} \frac{1-2p_i}{4\log\frac{1-p_i}{p_i}}, & \text{if } p_i \neq \frac{1}{2}; \\ \frac{1}{8}, & \text{if } p_i = \frac{1}{2}, \end{cases} \qquad \text{for} \quad p_i = \frac{\mathbb{E}\left[X_i\right] - a_i}{b_i - a_i}.$$

Indeed, we see that Hoeffding's inequality uses $c_i = \frac{1}{8}$ uniformly, while the expression given here goes to 0 as p_i approaches either 0 or 1. Berend and Kontorovich (2013) give an improved bound with $c_i = \frac{p(1-p)}{2}$ for $\frac{1}{2} \leq p_i \leq 1$.

4.3 Uniform Deviations

We have seen concentration inequalities bounding the probability of receiving undesired samples; in fact, if we showed that for any fixed parameter the log-likelihood function (and therefore a function only of worker label matrices) had bounded dependence on each of the entries in the worker matrices, then McDiarmid's inequality would exponentially upper bound the probability of receiving a bad worker matrix – one to which the log-likelihood function maps a value that is too far from its expectation. But note, this argument is applied only at one particular, though arbitrary, parameter setting. When considering a set of parameters, the set of bad worker matrices may be different for each, so we explicitly need to consider the probability of receiving a worker matrix to which the log-likehood

function evaluated at any of these parameters maps a value that is too far off. We treat here the problem of providing probabilistic guarantees bounding such deviations uniformly over a collection of functions.

We loosely follow Bousquet et al. (2004) for the treatment of this subject matter. Consider a class of measurable functions $\mathcal{F}$ from the domain $\mathcal{X}$, parametrized by some Θ:

$$\mathcal{F} = \{f_\theta : \mathcal{X} \to \mathbb{R} \mid \theta \in \Theta\}.$$

Furthermore, let X be a $(\mathcal{X}, \mu)$-valued random variable, for any μ measure on $\mathcal{X}$ Concentration inequalities in Section 4.2 are concerned with

$$f_\theta(X) - \mathbb{E}\left[f_\theta(X)\right], \tag{4.2}$$

for a fixed θ. Our quantity of interest is the uniform deviation

$$\sup_\theta f_\theta(X) - \mathbb{E}\left[f_\theta(X)\right], \tag{4.3}$$

which is an upper bound of Equation (4.2) for any $\theta \in \Theta$. First consider the case that Θ has finite cardinality N; we show how to control Equation (4.3) in terms of Equation (4.2). For every $\theta \in \Theta$, let

$$C_\theta = \{x \in \mathcal{X} : f_\theta(x) - \mathbb{E}\left[f_\theta(x)\right] > \varepsilon\},$$

the bad set. Notice that using these we can qualify the bad set C for Equation (4.3): if $x \in C_\theta$ for any θ then $x \in C$ must also be the case. That is, $C \subseteq \cup_\theta C_\theta$, and in fact needs not be larger. By the union bound

$$\mathbb{P}(C) = \mathbb{P}(\cup_\theta C_\theta) \leq \sum_\theta \mathbb{P}(C_\theta),$$

whose terms are individually controlled by concentration inequalities, giving $\mathbb{P}(C_\theta) \leq \delta_\theta$. Often we simply use just one $\delta \geq \delta_\theta$. Then

$$\mathbb{P}\left(\{x \in \mathcal{X} : \max_\theta f_\theta(x) - \mathbb{E}\left[f_\theta(x)\right] > \varepsilon\}\right) \leq N\delta.$$

When Θ is uncountably infinite this technique is bound to fail, though. In order to provide bounds for such situations one commonly selects a finite number of "representative" parameters and argues through those. We introduce a way of doing so in the following Section.

4.4 Covering Numbers

We introduce covering numbers in order to talk about all points of a pseudometric space through a small set of proxy elements. Results, unless otherwise stated, are from Bartlett (2013). Towards a precise definition suppose (S, d) is a pseudometric space. An ε-cover for (S, d) is a subset $\mathcal{C} \subseteq S$ such that for all $s \in S$ there exists $c \in \rfloor$ with $d(s, c) \leq \varepsilon$. The ε-covering number of (S, d) is defined as

$$N(\varepsilon, S, d) \doteq \min\{\mathcal{C} : \mathcal{C} \text{ is an } \varepsilon\text{-cover of } S\},$$

that is, the size of the ε-cover with the smallest size. The following lemma showcases how an ε-cover may be used to bound uniform deviations over function classes by the uniform deviation over just its cover.

Lemma 5. *Let Θ be a set, and $\mathcal{F}$ a class of bounded and measurable functions $f : \mathcal{X} \to \mathbb{R}$ parametrised by it. We define a pseudo-metric on Θ by*

$$d(\theta, \theta') = \|f_\theta - f_{\theta'}\|_\infty .$$

Let $\mathcal{C}$ be an $\varepsilon/3$-cover of (Θ, d). As before, we let X be a $(\mathcal{X}, \mu)$-valued random variable, for any μ measure on $\mathcal{X}$. Then

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |f_\theta(X) - \mathbb{E}[f_\theta(X)]| > \varepsilon\right) \leq \mathbb{P}\left(\sup_{\theta\in\mathcal{C}} |f_\theta(X) - \mathbb{E}[f_\theta(X)]| > \frac{\varepsilon}{3}\right).$$

Proof of Lemma 5. The proof is based solely on passing to the parameters in the cover, and controlling the approximation error. Take arbitrary $\theta \in \Theta$. By the definition of $\mathcal{C}$, there is a $\theta' \in \mathcal{C}$ such that $d(\theta, \theta') < \varepsilon/3$. Thus

$$\begin{aligned} |f_\theta(X) - \mathbb{E}[f_\theta(X)]| &\leq |f_\theta(X) - f_{\theta'}(X)| + |f_{\theta'}(X) - \mathbb{E}[f_{\theta'}(X)]| + |\mathbb{E}[f_{\theta'}(X)] - \mathbb{E}[f_\theta(X)]| \\ &\leq |f_{\theta'}(X) - \mathbb{E}[f_{\theta'}(X)]| + \frac{2}{3}\varepsilon. \end{aligned}$$

The first inequality is seen by introducing the extra terms, then applying the triangle inequality. In the second, we use the definition of the cover twice, along with noting that by the linearity of the expectation

$$|\mathbb{E}[f_{\theta'}(X)] - \mathbb{E}[f_\theta(X)]| = |\mathbb{E}[f_{\theta'}(X) - f_\theta(X)]| \leq \mathbb{E}[|f_{\theta'}(X) - f_\theta(X)|] \leq \mathbb{E}[\varepsilon/3].$$

We complete the proof by noting

$$\left(\varepsilon < \sup_{\theta\in\Theta} |f_\theta(X) - \mathbb{E}[f_\theta(X)]|\right) \quad \Rightarrow \quad \left(\varepsilon - \frac{2}{3}\varepsilon < \max_{\theta\in\mathcal{C}} |f_{\theta'}(X) - \mathbb{E}[f_{\theta'}(X)]|\right). \qquad \square$$

Stronger results can be obtained by considering covers of various sizes – on a high level, some might fit our space better than others. A technique called chaining automatically selects the appropriate scales and often leads to slightly improved bounds. We have not utilised this technique in this thesis, and therefore we instead turn our attention to tools that simplify the calculation of the covering number of specific spaces.

Lemma 6. *For a pseudometric space (Θ, d_Θ) and a class of functions parametrised by it, $\mathcal{F} = \{f_\theta\}$, equipped with its own pseudometric $d_\mathcal{F}$, if there is some L s.t. for any $\theta, \theta' \in \Theta$*

$$d_\mathcal{F}(f_\theta, f_{\theta'}) \leq L \cdot d_\Theta(\theta, \theta'),$$

that is, if the map $\theta \mapsto f_\theta$ is L-Lipschitz w.r.t the appropriate pseudometric spaces, then

$$N(\varepsilon, \mathcal{F}, d_\mathcal{F}) \leq N(\varepsilon/L, \Theta, d_\Theta).$$

Lemma 7. *Consider any norm $\|\cdot\|$ on $\mathbb{R}^d$, and let B be the unit ball. Then*

$$\frac{1}{\varepsilon^d} \leq N(\varepsilon, B, \|\cdot\|) \leq \left(\frac{2}{\varepsilon} + 1\right)^d.$$

Chapter 5

Results

Our main result is a finite sample bound on the loss suffered by the maximum likelihood estimate. In Section 5.1 we give a sequence of steps leading to this bound. While the final result is technically a conjecture, we structure the argument in a way to illustrate how a similar, though somewhat weaker and asymptotic statement about the error can be rigorously proven. In Section 5.2 we collect additional facts that may provide more insight into the problem. Proofs will appear in a designated chapter, 7.

5.1 A Convergence Rate for Label Inference

Recall that in the one-coin model we are working with a parametrized – by the true labels and skills – family of distributions; therefore it is natural to infer the labels from the observed worker response matrix Y using the maximum likelihood estimator. In this section we develop a bound for the loss suffered by this estimate.

In order to make a high probability assertions about the proximity of the MLE, $\hat{\theta}$, and the generating parameters, θ^*, we will pass to the log-likelihood function and study its properties. Here we use θ in place of the pair of parameters (s, y). We denote the log-likelihood function of a worker label matrix A by $\mathcal{L}(\theta \mid A)$, the random function associated to it by $\mathcal{L}(\theta \mid Y)$ or $\mathcal{L}(\theta)$ and finally its expected value by $\overline{\mathcal{L}}(\theta)$. First we work towards showing that the maximizer $\overline{\theta}$ of the expected log-likelihood is often close to $\hat{\theta}$. For this purpose, we study how $\mathcal{L}(\cdot)$ concentrates around $\overline{\mathcal{L}}(\cdot)$. We find that on most samples we their deviation can be bounded uniformly over the full domain under consideration. This guarantees that their maxima must also be close on the same samples. Finally, due

to the shape of the log-likehood function, we find the maximizers themselves are close, then. All that remains to show is that $\overline{\theta}$ is close to θ^*; we will find that they coincide. A complication that arises is that the bound on the loss derived from this argument is weak. To overcome this issue we do not use the label estimates directly, but plug the skill estimates into the optimal decision rule and analyse the error of its output. Details of the precise and quantitative results are to follow.

5.1.1 The Log-Likelihood Functions and Their Maximizers

Let Y be a binary $w \times t$ random matrix that follows the one-coin model (3.1) with the true skills and labels being $s \in [-1,1]^w$ and $y \in \{\pm 1\}^t$, respectively. Let $p(\cdot; s, y)$ denote the corresponding probability mass function over $\{\pm 1\}^{w \times t}$. Then,

$$p\left(x; s, y\right) = \mathbb{P}\left(Y = x\right) = \prod_{j \in [t]} \prod_{i \in [w]} p\left(x_{ij}; s_i, y_j\right) = \prod_{j \in [t]} \prod_{i \in [w]} \left(\frac{1+s_i}{2}\right)^{\mathbb{I}\{x_{ij} = y_j\}} \left(\frac{1-s_i}{2}\right)^{\mathbb{I}\{x_{ij} \neq y_j\}} .$$

Therefore, the log-likelihood function (LLF), with normalization that will be convenient later, becomes

$$\begin{aligned}
\mathcal{L}\left(s, y \,|\, x\right) &= \frac{1}{wt} \sum_{j \in [t]} \sum_{i \in [w]} \left[\mathbb{I}\left\{x_{ij} = y_j\right\} \log \frac{1+s_i}{2} + \mathbb{I}\left\{x_{ij} \neq y_j\right\} \log \frac{1-s_i}{2} \right] \\
&= \frac{1}{wt} \sum_{j \in [t]} \sum_{i \in [w]} \left[\frac{1+x_{ij}y_j}{2} \log \frac{1+s_i}{2} + \frac{1-x_{ij}y_j}{2} \log \frac{1-s_i}{2} \right] \\
&= \frac{1}{2}\frac{1}{w} \sum_{i \in [w]} \left[\log \frac{1-s_i^2}{4} \right] + \frac{1}{2}\frac{1}{wt} \sum_{i,j} \left[x_{ij} y_j \log \frac{1+s_i}{1-s_i} \right] .
\end{aligned}$$

A maximum likelihood estimator (MLE) maps $x \in \{\pm 1\}^{w \times t}$ to any global maximizer of this function:

$$f^{MLE}(x) \in \operatorname*{arg\,max}_{(s,y) \in [-1,1]^w \times \{\pm 1\}^t} \mathcal{L}\left(s, y \,|\, x\right) .$$

Indeed, there are at least two such estimates. As noted earlier, the parametrization of the model is not identifiable: in particular (s, y) and $(-s, -y)$ define the same distribution for any s, y. By the symmetry breaking assumption, Equation (3.4), however, we will take the one where the sum of the skills in vector s is positive.

For the rest of the argument we assume that Y follows the one-coin model with skills and labels given by $s_* \in [-1,1]^w$ and $y_* \in \{\pm 1\}^t$, respectively, and we will study the

behaviour of $(\hat{s}, \hat{y}) = f^{MLE}(Y)$. Note that $\hat{s}, \hat{y}$ are random, as Y is random. Recall from Section 3.1 that in this case we may write the worker reported labels in terms of the true label and the worker correctness matrix T through $T_{ij} = Y_{ij} y_j^*$. For the purposes of the analysis and to simplify the presentation, we let $y_j^* = 1$ for all $1 \in [t]$. Then the LLF evaluated at $x = Y$ becomes

$$\mathcal{L}(s, y \,|\, T) = \frac{1}{2}\frac{1}{w} \sum_{i \in [w]} \left[\log \frac{1 - s_i^2}{4} \right] + \frac{1}{2}\frac{1}{wt} \sum_{i,j} \left[T_{ij} y_j \log \frac{1 + s_i}{1 - s_i} \right]. \tag{5.1}$$

In order to analyse the estimates derived from "typical" correctness matrices T, we study the expected log-likelihood function $\overline{\mathcal{L}}$ defined by $\overline{\mathcal{L}}(s, y) = \mathbb{E}[\mathcal{L}(s, y \,|\, Y)]$, noting that one expects $\mathcal{L}(\cdot, \cdot \,|\, Y)$ to be "close" to $\overline{\mathcal{L}}(\cdot, \cdot)$ under appropriate conditions as the quantity of the data increases. Taking the expectation of $\mathcal{L}(s, y \,|\, Y)$, from our previous assumption on y^*, we compute

$$\begin{aligned} \overline{\mathcal{L}}(s, y) &= \mathbb{E}\left[\frac{1}{2}\frac{1}{w} \sum_{i \in [w]} \left[\log \frac{1 - s_i^2}{4} \right] + \frac{1}{2}\frac{1}{wt} \sum_{i,j} \left[T_{ij} y_j \log \frac{1 + s_i}{1 - s_i} \right] \right] \\ &= \frac{1}{2}\left[\frac{1}{w} \sum_{i \in [w]} \log \frac{1 - s_i^2}{4} \right] + \frac{1}{2}\left[\frac{1}{t} \sum_{j \in [t]} y_j \right] \left[\frac{1}{w} \sum_{i} s_i^* \log \frac{1 + s_i}{1 - s_i} \right]. \end{aligned} \tag{5.2}$$

We used the linearity of expectations, and noticed the second sum becomes separable. Let

$$(\bar{s}, \bar{y}) \in \operatorname*{arg\,max}_{(s,y) \in [-1,1]^w \times \{\pm 1\}^t} \overline{\mathcal{L}}(s, y).$$

We remind the reader that $(\bar{s}, \bar{y})$ is a deterministic element of the parameter space. A simple direct calculation (found in Section 7.1) gives the following fact:

Fact 1. *The maximizers of the expected LLF coincide with the true generating parameters and their negations, that is,*

$$\{(\bar{s}, \bar{y}), (-\bar{s}, -\bar{y})\} = \{(s_*, y_*), (-s_*, -y_*)\}.$$

Notation for the log-weights. As $\log \frac{1+s}{1-s}$, which we simply call the weight function, appears consistently in the remaining of the document we use the shorthand v for it, or $v(s)$ when its dependence on s requires attention. We further abuse notation by allowing $v(\cdot)$ to have domain $[-1, +1]$, or $[-1, +1]^w$, whichever is required in the given context. For

the latter, the weight function is applied element-wise. As this function is injective in its domain – with the range of extended reals $\bar{\mathbb{R}}$ – we can work equivalently in the s (skill) or v (weight) domains. The inverse of weight function is, in fact, the sigmoid function with range $[-1, +1]$. We translate the s_*, $\hat{s}_i$, etc. notation to the weight domain, too, as $v_*, \hat{v}_i, \ldots$. See Section 7.7 for a detailed treatment of these function.

5.1.2 The Convergence of the Log-Likelihood to its Expectation

In this step we analyse the rate at which the (random) LLF approaches its expected value. The distance between these functions will be measured in the supremum norm $\|\cdot\|_\infty$, that is, as the "largest" difference in function values for any point in the parameter space. This so called uniform deviation bound will allow us to guarantee convergence regardless of what the true parameters are. Mathematically speaking, we will find some measurable function f such that

$$\mathbb{P}\left(\sup_{(s,y)\in[-1,1]^w\times\{\pm1\}^t} |\mathcal{L}(s,y\,|\,T) - \overline{\mathcal{L}}(s,y)\,| < f_{s_*}(\delta,w,t)\right) > 1-\delta.$$

There are a few difficulties. First, the deviation

$$\Delta = \sup_{(s,y)\in[-1,1]^w\times\{\pm1\}^t} |\mathcal{L}(s,y\,|\,T) - \overline{\mathcal{L}}(s,y)\,|$$

is difficult to control as the magnitudes of the derivatives of both $\mathcal{L}(s,y)$ and $\overline{\mathcal{L}}(s,y)$ diverge to ∞ if for some $i \in [w]$, $|s_i|$ approaches one. Hence, they are very sensitive in this region which makes it hard to control their difference. We take care of this, by restricting the range of s_i (for each i) to $S_\lambda \doteq [-1+\lambda, 1-\lambda]$. The next issue is that the deviation Δ might not be a measurable function, making the expression in the above displayed equation ill-formed. In our case, however, Δ can be shown to be measurable based on a result stated in Appendix C of the book of Pollard (1984).

To derive the function f_{s_*}, we employ the concept of covering numbers. This will allow us to bound the deviation over the whole domain through a finite number -the covering number of our space, to be precise- of proxy elements in it (cf. Lemma 5). For each such element we calculate point-wise tail bounds on the deviation.

Lemma 8 (Concentration of the LLF at a fixed parameter)**.** *For $T \in \{\pm1\}^{w\times t}$ generated according to true skills $s_* \in [-1,1]^w$, for any $s \in [-1+\lambda, 1-\lambda]^w$, $y \in \{\pm1\}^t$, we have*

$$\mathbb{P}\left(\left|\overline{\mathcal{L}}(s,y) - \mathcal{L}(s,y\,|\,T)\right| \geq \varepsilon\right) \leq 2\exp\left(-\frac{w^2 t\varepsilon^2}{\log^2(2\lambda^{-1}-1)\sum_i \frac{s_i^*}{v_i^*}}\right).$$

Note that the term $\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}$ is bounded by $\frac{1}{2}$ and decreases to 0 as the quality of workers goes up, strengthening the bound. This refinement is missed by traditional concentration inequalities such as that of Hoeffding or McDiarmid; we use an inequality due to Kearns and Saul (1998) for the proof (cf. Lemma 3). For the covering numbers, we have the following result:

Lemma 9 (The covering number of our space)**.** *Consider the pseudometric on our parameter space* $\Theta = [-1+\lambda, 1-\lambda]^w \times \{\pm 1\}^t$ *given by*

$$d((s,y),(s',y')) = \left\| \mathcal{L}\left(s, y \,|\, \cdot\right) - \mathcal{L}\left(s', y' \,|\, \cdot\right) \right\|_\infty .$$

Let $N(\varepsilon, \Theta, d)$ *denote the* ε*-covering number of* Θ*. Then we have*

$$N(\varepsilon/3, \Theta, d) \leq \left(\frac{6\left\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\right\}}{\varepsilon} + 1 \right)^{w+t} .$$

Combining these results with the uniform deviation bound of Lemma 5, we arrive at

$$\begin{aligned}
\mathbb{P}\left(\sup_{\theta\in\Theta} \left| \mathcal{L}\left(\theta \,|\, T\right) - \overline{\mathcal{L}}\left(\theta\right) \right| \geq \varepsilon \right) &\leq \mathbb{P}\left(\max_{\theta\in\mathcal{C}} \left| \mathcal{L}\left(\theta \,|\, T\right) - \overline{\mathcal{L}}\left(\theta\right) \right| \geq \frac{\varepsilon}{3} \right) \\
&\leq \sum_{j\in\mathcal{C}} \mathbb{P}\left(\left| \mathcal{L}\left(\theta \,|\, T\right) - \overline{\mathcal{L}}\left(\theta\right) \right| \geq \frac{\varepsilon}{3} \right) \\
&\leq |\mathcal{C}| 2 \exp\left(- \frac{w^2 t \varepsilon^2}{9 \log^2(2\lambda^{-1}-1) \sum_i \frac{s_i^*}{v_i^*}} \right) \\
&\leq 6\left(\frac{2\left\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\right\}}{\varepsilon} + 1 \right)^{w+t} \cdot \exp\left(- \frac{w^2 t \varepsilon^2}{9 \log^2(2\lambda^{-1}-1) \sum_i \frac{s_i^*}{v_i^*}} \right) .
\end{aligned}$$

It is often informative to invert this bound so that the guarantee on the maximum size of the deviation is expressed in terms of the probability of a bad sample, $\delta > 0$. Doing so leads to the following lemma.

Lemma 10 (High probability uniform deviation bound)**.** *Fix* $s_* \in [-1,1]^w$ *and let*

$$\varepsilon_{s_*}(\delta, w, t) \doteq 3\log(2\lambda^{-1}-1) \sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}} \sqrt{\frac{1}{2wt}\left[(w+t)\log C + \log(2/\delta) \right]} , \tag{5.3}$$

where C is a problem dependent constant that is polynomial in w, t (the exact expression for C is given in (5.4)). For $T \in \{\pm 1\}^{w \times t}$ generated according to true skills $s_* \in [-1,1]^w$, $\Theta = S_\lambda^w \times \{\pm 1\}^t$, for any $\delta > 0$, we guarantee

$$\mathbb{P}\left(\sup_{(s,y)\in\Theta} \left| \mathcal{L}\left(s,y\right) - \overline{\mathcal{L}}\left(s,y\right) \right| < \varepsilon_{s_*}(\delta, w, t) \right) > 1 - \delta\,.$$

The value of C mentioned in this lemma is

$$C = \frac{6\sqrt{2wt}\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\}}{3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\log(2/\delta)}} + 1\,. \tag{5.4}$$

We can see that this quantity does not increase fast with w, t increasing, λ decreasing, and in fact decreases with δ decreasing to 0. Considering that their contribution to ε_{s_*} is through a logarithm, this term is almost negligible. It is only $\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}$ that can easily make C explode, when a true skill approaches ± 1. More precisely, by Fact 3 (shown in the supplementary material), $\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*} \geq \frac{1}{2+\overline{\Phi}}$, thus, for small enough λ and δ,

$$C = O\Big((1+\tfrac{1}{\lambda})\sqrt{wt(1+\overline{\Phi})}\Big)\,.$$

The proofs of these claims appear in Section 7.2.

5.1.3 The Proximity of the Maximizing Parameters

We have proven that for most worker label matrices generated according to the one-coin model the empirical and expected LLFs become close when the quantity of data is large – in fact we quantified this relationship. Now we show that this is enough to conclude that each maximizer $\hat{\theta} = (\hat{s}, \hat{y})$ of the LLF over the domain $\Theta = S_\lambda^w \times \{\pm 1\}^t$ associated to such a worker label matrix is close to a maximizer $\bar{\theta} = (\bar{s}, \bar{y})$ of the expected LLF. By Lemma 13, applied to the empirical and expected LLFs, assuming that $\bar{\theta} \in \Theta$,

$$|\overline{\mathcal{L}}(\hat{\theta}) - \overline{\mathcal{L}}(\bar{\theta})| \leq 2 \sup_{\theta\in\Theta} \left| \mathcal{L}\left(\theta\right) - \overline{\mathcal{L}}\left(\theta\right) \right|\,, \tag{5.5}$$

that is, for some bounded uniform deviation of the empirical from the expected LLF we get a bounded difference in the values of the expected LLF at the two maximizing parameters.

This in turn will lead to bounds on how far these parameters (the skills and labels) can be. To paraphrase, we find an easy to understand superset of

$$\{\theta \in \Theta \mid \overline{\mathcal{L}}(\theta) \geq \overline{\mathcal{L}}\left(\bar{\theta}\right) - \varepsilon\} = \overline{\mathcal{L}}^{-1}(\overline{\mathcal{L}}\left(\bar{\theta}\right) - \varepsilon),$$

which (implicitly) bounds the error on the label and skill estimates.

Recall that y only interacts with the expected LLF values through its average (cf. Equation (5.2)). It is an option to bound the number of mistakes by finding $\check{y} \in \{\pm 1\}^t$ with smallest absolute value satisfying

$$(s, \check{y}) \in \overline{\mathcal{L}}^{-1}(\overline{\mathcal{L}}\left(\bar{\theta}\right) - \varepsilon)$$

for some s, though we found this leads to weak bounds.

Instead, we concentrate on bounding the error in the skill, or rather, the weight estimates as a function of the deviation between $\overline{\mathcal{L}}(s, y) - \overline{\mathcal{L}}(\bar{s}, \bar{y}) = \overline{\mathcal{L}}(s, y) - \overline{\mathcal{L}}(s_*, \mathbb{1})$ (recalling that $\bar{y} = \mathbb{1}$ for the sake of analysis, and that the corresponding $\bar{s}$ is s_*). We remind the reader that the weight associated to a skill s is $v = \log \frac{1+s}{1-s}$, and that it corresponds to the weight that a worker with the given skill receives in the optimal decision rule. We present two bounds; the first is asymptotic, the second relies on an unproven analytic conjecture, also stated below. Both rely on the same real-analysis argument, given in Section 7.3.

Theorem 2. *Fix any $0 < \gamma < 1$, $s_*, s \in [-1, 1]^w$, $y \in \{\pm 1\}^t$ and consider $\overline{\mathcal{L}} : [-1, 1]^w \times \{\pm 1\}^t \to \mathbb{R}$. There exists $\varepsilon_0 > 0$ such that if*

$$\varepsilon \doteq \overline{\mathcal{L}}(s_*, 1) - \overline{\mathcal{L}}(s, y) < \varepsilon_0$$

then the weight estimates enjoy

$$\|v(s) - \chi v(s_*)\|_1 \leq \frac{2}{\gamma} \sqrt{2w\varepsilon \sum_i \frac{1}{1 - (s_i^*)^2}}$$

for $\chi = \operatorname{sgn}\left(\sum_j y_j\right)$.

Conjecture 1. *For any $0 < \lambda < 0.0067$, $s_* \in [-1, 1]$, $s \in S_\lambda$ we have*

$$\log \frac{1 - s_*^2}{1 - s^2} + s_*(v_* - v) \geq \frac{1 - s_*^2}{2 \log(1/\lambda)} (v_* - v)^2 .$$

Although the conjecture is supported by extensive numerical calculations, we have been unable to show it analytically. See Appendix A for a justification of why the conjecture is believable.

Theorem 3. *Assuming Conjecture 1 holds, for any* $0 < \lambda < 0.0067$, $s_* \in [-1,1]^w$, $s \in S_\lambda^w$, $y \in \{\pm 1\}^t$, *letting*

$$\varepsilon \doteq \overline{\mathcal{L}}\left(s_*, 1\right) - \overline{\mathcal{L}}\left(s, y\right) \ ,$$

the weight estimates are bounded as

$$\left\| v(s) - \chi v(s_*) \right\|_1 \leq 2\sqrt{w\varepsilon \log\left(\frac{1}{\lambda}\right) \sum_i \frac{1}{1-(s_i^*)^2}}$$

for $\chi = \operatorname{sgn}\left(\sum_j y_j\right)$.

The conditions are quite mild. We see that we recover the weights or their negations depending on the sign of the average label y. Again, when "running" the MLE, we make sure to choose the label estimate corresponding to the skills whose average is positive – in accordance with the assumption that workers on average label correctly (cf. Equation (3.4)).

It may be more natural to interpret the results in terms of the normalized ℓ_1 norm. The conclusion of Theorem 3, for example, is

$$\frac{1}{w}\left\| v(s) - v(s_*) \right\|_1 \leq 2\sqrt{\varepsilon \log\left(\frac{1}{\lambda}\right)\left[\frac{1}{w}\sum_i \frac{1}{1-(s_i^*)^2}\right]} \ .$$

We clearly see that the average proximity is governed by a measure of the quality of the crowd, $\sqrt{\frac{1}{w}\sum_i \frac{1}{1-(s_i^*)^2}}$, and $\sqrt{\varepsilon \log(1/\lambda)}$. We find that the more skilled the workers are, the harder it is to approximate their weights. This is not surprising: the weights of highly qualified workers are extremely sensitive to changes in the skill. The term $\log(1/\lambda)$ is quite small, even if we tie $1/\lambda$ to some polynomial function of the data size. Recall, the ε appearing here is related to the uniform deviation calculated in the previous section, through Lemma 13. Next, we investigate what errors we make when using approximate weights, of any source, in the optimal decision rule. Our argument will climax in Section 5.1.5 when we combine all these results.

5.1.4 From Approximate Weights to Labels

In this section, given some weights $\hat{v} \approx v_*$, we want to bound the probability of returning an incorrect label by using weighted majority voting for a single task. (Note that the same argument applies if $\hat{v} \approx -v_*$, except then the incorrect label would actually be $+1$. Without loss of generality we consider the "positive" case, that is, $\hat{v} \approx v_*$.) Recall, the v-weighted majority voting decision rule is

$$\begin{aligned} f^v : \{\pm 1\}^t &\to \{\pm 1\} \\ x &\mapsto \operatorname{sgn}\left(x^\top v\right), \end{aligned} \tag{5.6}$$

which we will apply to the labels Y provided by workers for an instance (thus, in this section Y and T are column vectors corresponding to a single task). Recall, under the assumption $y_* = \mathbb{1}$ the decision rule becomes $\operatorname{sgn}\left(v^\top T\right)$, and it makes a mistake if it is not 1. We relate the event that f^v makes a mistake to the optimal decision rule f^{v_*} making an error. As we will apply this argument to our weight estimates $\hat{v}$, in the analysis that follows $\hat{v}$ denotes a random vector. Now, for any $\varepsilon > 0$

$$\begin{aligned} \mathbb{I}\left\{\operatorname{sgn}\left(\hat{v}^\top T\right) \neq 1\right\} &\leq \mathbb{I}\left\{\hat{v}^\top T \leq 0\right\} = \mathbb{I}\left\{v_*^\top T \leq (v_* - \hat{v})^\top T\right\} \\ &\leq \mathbb{I}\left\{v_*^\top T \leq (v_* - \hat{v})^\top T,\, (v_* - \hat{v})^\top T \leq \varepsilon\right\} \\ &\quad + \mathbb{I}\left\{v_*^\top T \leq (v_* - \hat{v})^\top T,\, (v_* - \hat{v})^\top T > \varepsilon\right\} \\ &\leq \mathbb{I}\left\{v_*^\top T \leq \varepsilon\right\} + \mathbb{I}\left\{(v_* - \hat{v})^\top T > \varepsilon\right\} \\ &\leq \mathbb{I}\left\{v_*^\top T \leq \varepsilon\right\} + \mathbb{I}\left\{\|v_* - \hat{v}\|_1 > \varepsilon\right\}, \end{aligned} \tag{5.7}$$

where in the last step we used that $|T_i| \leq 1$. Intuitively, the bound states we make a mistake using $\hat{v}$ if the best weights are too close to (or are) making a mistake, or our weight estimates are too far off from the best ones. The following lemma, utilising Kearns and Saul's inequality in its proof (Section 7.4.1), bounds the probability of the former event. Recall that Φ denotes the total committee potential of our crowd, which fully captures its labelling power (cf. Section 3.4.1).

Lemma 11. *If $\Phi \neq 0$, for any $\varepsilon > 0$, $r > 0$*

$$\mathbb{P}\left(v_*^\top T \leq \varepsilon\right) \leq \exp\left\{r\Phi(r - (1 - \tfrac{\varepsilon}{\Phi}))\right\},$$

and assuming

$$\varepsilon \leq \Phi, \tag{5.8}$$

we have

$$\mathbb{P}\left(v_*^\top T \leq \varepsilon\right) \leq \exp\left\{-\frac{\Phi}{4}\left(1-\frac{\varepsilon}{\Phi}\right)^2\right\}.$$

We turn our attention to the second event, that the approximate weights are too far from the true weights. The work we have done so far has been towards bounding the probability of this event: Lemma 10 quantifies the uniform deviations in the LLF, which, combined with Theorem 2 or Theorem 3, yield high probability bounds on the weight approximation error. Without concern for the details, we observe that the bounds will be of the form

$$\mathbb{P}\left(\|\hat{v}-v_*\|_1 > C_1\sqrt[4]{C_2+\log(1/\delta)}\right) \leq \delta \tag{5.9}$$

for some $C_1, C_2 > 0$ problem dependent constants where $0 < \delta \leq 1$. Theorem 3 allows δ to range over $0 < \delta \leq 1$, but in order to apply Theorem 2 we require $\delta \geq \delta_0$ for some δ_0 dependent on the remaining parameters. We remark that δ_0 decreases with the size of the problem (while other parameters are held constant) and becomes 0 when the parameters cross a certain, problem dependent threshold.

What remains is to choose a way to distribute a prescribed total error between the two upper bounding events, that is, to choose an ε to apply Equation (5.7) with. Now, one may optimize the value of ε to minimize an upper bound on the expectation of the right-hand side of Equation (5.7), but we strive for a simple presentation instead: we relate the exponents of the two mistake bounds so that their sum is simply controlled from above. This yields the following general result on the loss, ℓ, of weighted majority voting with any approximate weights that have high probability bounds of the form outlined above. We bring to the reader's attention that the weights are allowed to depend on T in this result.

Theorem 4. *Let v_* be the true weights, $\hat{v}$ estimates; let C_1, C_2 be some positive constants such that for all $0 < \delta < 1$ we are guaranteed*

$$\mathbb{P}\left(\|\hat{v}-v_*\|_1 > C_1\sqrt[4]{C_2+\log(1/\delta)}\right) \leq \delta\,.$$

Furthermore, define

$$t_1 = \frac{\Phi}{4}\left(1+\frac{C_1\sqrt[4]{C_2}}{2\sqrt{\Phi}}\right)^{-2}, \qquad t_2 = \frac{\Phi}{4}\left(1+\frac{C_1}{2\sqrt{\Phi}}\right)^{-2}.$$

Then if C_2, $\log(2/\delta) \geq 1$ the probability of the $\hat{v}$-weighted majority algorithm making an error is bounded as

$$\mathbb{P}\left(\operatorname{sgn}\left(\hat{v}^\top T\right) \neq 1\right) \leq 2\exp\left\{-\min(t_1, t_1^2)\right\}, \tag{5.10a}$$

furthermore,

$$\mathbb{P}\left(\operatorname{sgn}\left(\hat{v}^\top T\right) \neq 1\right) \leq 2\exp\left\{-\min(t_2, t_2^2 + C_2)\right\} \tag{5.10b}$$

always holds.

We see that for the error of $\hat{v}$ to have a negligible effect on the reconstruction error one needs C_1 (and C_2) to be small compared to the total committee potential. Finally, we remark that using weights $\hat{v}$ close to $-v_*$ we get the same bounds on $\mathbb{P}\left(\operatorname{sgn}\left(\hat{v}^\top T\right) \neq -1\right)$. See Section 7.4.2 for the proof of the theorem.

5.1.5 A Mistake Bound

In this section we simply combine earlier results for mistake bounds. We will bound the probability of mislabelling a randomly chosen task, our proposed loss. Note that here, by "mislabelling" a task, we mean we allow the consistent flipping of all labels and measure our error against the better one; the corresponding precise definition is Equation (3.5). We derive a finite sample bound (based on Conjecture 1). We will need the problem specific constant

$$C_2 = (w+t)\log\left(\frac{6\sqrt{2wt}\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\}}{3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\log(2/\delta)}} + 1\right) + \log(2)\,.$$

Theorem 5 (Finite sample mistake bound)**.** *Let $(\hat{s}, \hat{y})$ be the ML estimate over $\Theta = S_\lambda^w \times \{\pm 1\}^t$, $\hat{v} = v(\hat{s})$, $\tilde{y}_j = \operatorname{sgn}\left(\hat{v}^\top Y_{:,j}\right)$ be the label obtained using weighted majority voting with the random weights $\hat{v}$. Fix any $0 < \lambda < 0.0067$, $s_* \in S_\lambda^w$. Letting*

$$r \doteq \log(2/\lambda)\sqrt{6\sqrt{\frac{w}{2t}}\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\left(\frac{1}{w}\sum_i \frac{1}{1-(s_i^*)^2}\right)\overline{\Phi}^{-1}}\,,$$

and given that Conjecture 1 holds, for any $j \in [t]$,

$$\mathbb{P}\left(y_j \neq \tilde{y}_j\right) \leq 2\exp\left\{-\min\left(\tfrac{\Phi}{4}(1+r)^{-2},\ (\tfrac{\Phi}{4}(1+r)^{-2})^2 + C_2\right)\right\}\,.$$

The proof can be found in Section 7.5. Recall, the oracle predictor's loss (cf. Section 3.4.1) is $\exp(-\Phi)$ up to a constant in the exponent. Our bound is "diluted" by $(1+r)^{-2}$, therefore the closer r is to 0, the closer the guarantee on the loss suffered by

the ML estimates is to that of the oracle predictor. In order to understand the bounds we need to consider which term the minimum will select. Observe that if $C_2 \geq 1$, then the first term will automatically be smaller than the second term: if it were otherwise, it would require

$$\tfrac{\Phi}{4}(1+r)^{-2} \geq (\tfrac{\Phi}{4}(1+r)^{-2})^2 + C_2 \geq C_2 \geq 1\,,$$

but that means

$$1 \leq \tfrac{\Phi}{4}(1+r)^{-2} \leq (\tfrac{\Phi}{4}(1+r)^{-2})^2\,,$$

a contradiction. As $C_2 \leq w + t + \log(2)$, we can safely assume it is larger than one, and hence work with the upper bound as if it was

$$2\exp\left\{-\tfrac{\Phi}{4}(1+r)^{-2}\right\}\,.$$

First let us study the behaviour of the bound in the number of tasks we require labels for (while keeping other parameters constant). We see that for a large number of tasks we approximately have $r \sim t^{-1/4}$, and in turn for $r \to 0$: $(1+r)^{-2} \sim 1 - 2r$, yielding $(1+r)^{-2} \sim 1 - 2t^{-1/4}$. Therefore the error bound scales as

$$2\exp\{-\tfrac{\Phi}{4}(1-2t^{-1/4})\}$$

in t which approaches the error rate of the oracle as $t \to \infty$.

In order to study the effect of increasing the number of workers (while keeping others constant), notice that Φ is the total committee potential, that is, it is equal to $w\overline{\Phi}$. We are holding $\overline{\Phi}$ constant, while increasing w. Again, for large w, $r \sim w^{1/4}$, and in turn $\frac{w\overline{\Phi}}{4}(1+r)^{-2} \sim w^{1/2}$. We verify that the error of our estimation procedure decreases as workers are added, though the rate at which this happens is slower than that of any other method studied earlier, including uniform majority voting, whose exponents all scale linearly in the number of workers.

5.2 Additional Results

In this section we display any related results of some importance that do not directly contribute towards the mistake bound of the previous chapter.

5.2.1 Inference with Weights That are Independent of Labels

In Section 5.1.4 we showed a bound on the probability of mislabelling when using weight estimates $\hat{v}$. In the argument we had to allow $\hat{v}$ to be dependent on the random matrix T, as we would apply the result to weights calculated exactly based on the T. We now explicitly consider the case that the weights $\hat{v}$ are independent of the labels they are used on. Denote by

$$e(v) = \mathbb{P}\left(\operatorname{sgn}\left(v^\top T\right) \neq 1\right)$$

the probability that we make a mistake with the deterministic weight vector v. We can show

Lemma 12. *For $v \in \mathbb{R}^w$ fixed, the probability of error, $e(v)$, of the weighted majority rule on a new task is bounded by*

$$e(v) \leq \exp\left\{-\frac{1}{4}\frac{(v^\top s_*)^2}{\sum_{i=1}^w \frac{s_i^* v_i^2}{v_i^*}}\right\}. \tag{5.11}$$

The proof is presented in Section 7.6. For brevity, we denote the sum in the denominator by ρ, that is, $\rho(v) = \sum_{i=1}^w \frac{s_i^* v_i^2}{v_i^*}$. Note that when $v = v_*$, $\rho = v_*^\top s_*$, and $e(v_*) \leq e^{-\frac{1}{4} v_*^\top s_*}$, i.e., we get back Theorem 1(i) of Berend-Kontorovich. The uniform weighted majority voting algorithm is also of special interest. The above result directly yields the following corollary.

Corollary 2. *Let $\hat{y}_j$ denote the uniform majority estimate for task j. Then*

$$p_{\text{maj}} \doteq \mathbb{P}\left(\hat{y}_j \neq y_j^*\right) \leq \exp\left\{-\frac{1}{4}\left(\sum_{i=1}^w \frac{s_i^*}{v_i^*}\right)^{-1}\left(\sum_{i=1}^w s_i^*\right)^2\right\}. \tag{5.12}$$

5.2.2 Separation Criterion

We investigate the conditions under which workers can be distinguished from random guessers. This, if over half of the labels are estimated correctly, is equivalent to identifying the direction of every worker's bias correctly. To be precise, we give a condition on the size of the deviation of $\overline{\mathcal{L}}(s, y)$ from $\overline{\mathcal{L}}(\bar{s}, \bar{y})$ to guarantee this separation. (Recall $\bar{s} = s_*$, $\bar{y} = y_* = 1$ as per our assumption.)

We first characterize this problem for one worker. Recall the shape of the expected LLF function (cf. Equation (5.2)), which (in the weight domain) may be viewed on Figure 7.2.

Notice that some contour lines actually connect the regions of the positive and negative peaks. In such cases, even if $y > 0$ is guaranteed, the range of possible skills includes 0 in its interior.

These peaks are connected exactly if the region includes the low point of the saddle. Mathematically, the function value difference between the optimum and the low point $(0, 0)$ of the saddle is

$$\begin{aligned}\overline{\mathcal{L}}(s_*, 1) - \overline{\mathcal{L}}(0, 0) &= \frac{1}{2}\left[\log\frac{1-(s_*)^2}{4} - s_* \log\frac{1+s_*}{1-s_*}\right] - \frac{1}{2}\log\frac{1}{4} \\ &= \frac{1}{2}\left[\log(1-(s_*)^2) - s_* \log\frac{1+s_*}{1-s_*}\right] \\ &\doteq \frac{b(s_*)}{2},\end{aligned}$$

where $b(x)$ denotes the transformed negative binary entropy function shown on Figure 5.1. Note that it bounds the allowed deviation to be 0 for a worker of skill 0.

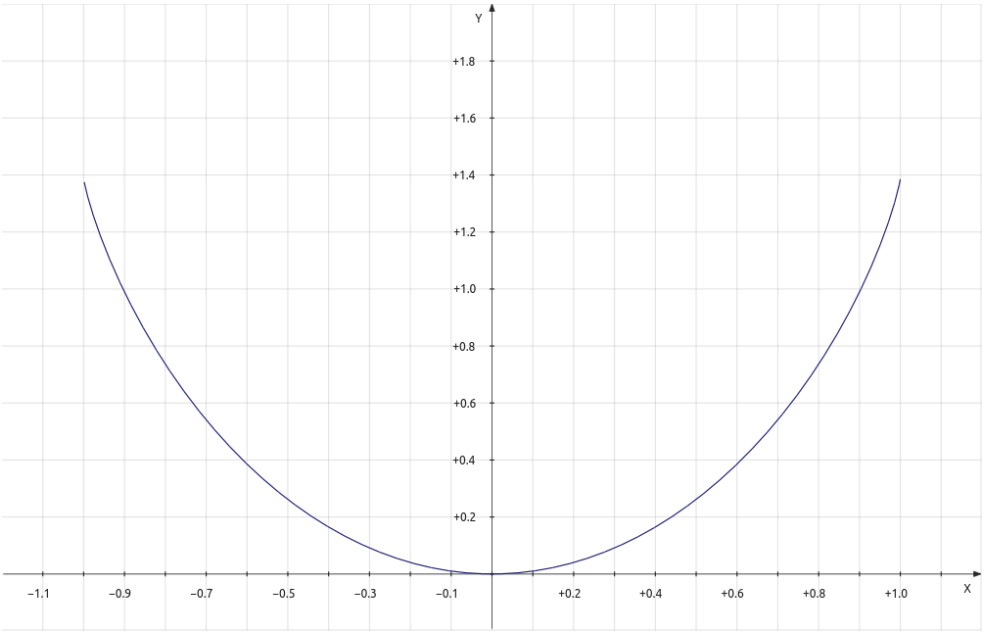

Figure 5.1: The transformed negative binary entropy acting as the Separation Criterion.

When considering multiple workers, we wish to guarantee that each one of them is distinguished from spammers. To guarantee this, $\varepsilon \leq b(s_i^*)/2$ is required for every worker i. Clearly this is also sufficient.

Chapter 6

Conclusion and Future Work

In this thesis, we studied various label aggregation or true label inference methods in the one-coin model — the simplest variant of the famed Dawid-Skene model. First, we gave an overview of the existing methods and their analysis. Next, we proceeded to contribute one of our own — claiming that the MLE associated to this model enjoys a low probability of mislabelling. We gave a proof of this claim, except for one well-justified, albeit not yet proven step regarding a function lower-bounding another.

The bound on the mislabelling error is presented in terms of the mislabelling error of the oracle predictor, the method that has knowledge of the skill of each worker. Therefore, in a way, it quantifies the additional error introduced by not having knowledge of these skills, and the possible looseness of our analysis. Furthermore, this presentation illustrates in what scenarios the performance of the ML estimates of the inference problem approach those of the oracle.

There are a number of steps in the argument where we may want to exert additional effort towards a possibly tighter analysis. In Section 5.1.2 instead of the uniform deviations we may be able to argue more locally near the extrema of the functions considered; in the same section, when using the covering numbers we may be able to use the so called chaining argument for a somewhat tighter uniform deviation bound; when bounding the probability that our weight estimates err on a task, we upper bound expected value of $\mathbb{I}\left\{(v_* - \hat{v})^\top T > \varepsilon\right\}$ by that of $\mathbb{I}\left\{\|v_* - \hat{v}\|_1 > \varepsilon\right\}$, while in reality T and $(v_* - \hat{v})$ may be related in such a way that the first quantity is significantly smaller. Furthermore, we may wish to replace the strategy currently used to distribute the error of mislabelling between the two terms in Equation (5.7): firstly, the event that the weights are not estimated well enough should not contribute towards the error of each individual task, rather we should

suffer it just once for the full reconstruction process; secondly, the distribution of the error may be optimized for a potentially better bound. Finally, the conjecture our result hinges on needs to be proven and we may also wish to carry the somewhat tighter asymptotic weight approximation result (Theorem 2) to a complete mistake bound.

It would also be educational to prove a (mimimax) lower bound on the performance of any inference algorithm in this setting. Additionally, we are very much interested in including task difficulties in the one-coin model, and have some preliminary results not presented here in that direction.

Chapter 7

Proofs

7.1 The Log-Likelihood Functions and Their Maximizers

Proof of Fact 1. We show that the set of maximizers of $\overline{\mathcal{L}}(s, y)$ is $\{(s_*, y_*), (-s_*, -y_*)\}$. By definition $\bar{y} \in \arg\max_y \overline{\mathcal{L}}(\bar{s}, y)$. As y only appears in the second term and $\overline{\mathcal{L}}$ is separable (cf. 5.2), we have

$$\bar{y}_j \in \arg\max_{y_j} \; y_j \sum_i s_i^* \log \frac{1+s_i}{1-s_i}.$$

By inspecting the right-hand side, we see that the maximum is achieved at $y_j = \operatorname{sgn}\left(\sum_i s_i^* \log \frac{1+s_i}{1-s_i}\right)$, which is actually constant over j, implying that either $\bar{y} = \mathbb{1}_t$ or $\bar{y} = -\mathbb{1}_t$. Consequently,

$$\overline{\mathcal{L}}(s, \bar{y}) = \frac{1}{2}\left(\frac{1}{w}\sum_{i\in[w]} \log\frac{1-s_i^2}{4} + \left|\frac{1}{w}\sum_{i\in[w]} s_i^* \log\frac{1+s_i}{1-s_i}\right|\right).$$

Next, we study $\bar{s}$, the skill vector that maximizes $\overline{\mathcal{L}}(s, \bar{y})$. In order to work with the absolute value appearing in this function, consider

$$l^+(s) = \frac{1}{2}\frac{1}{w}\sum_{i\in[w]}\left[\log\frac{1-s_i^2}{4} + s_i^* \log\frac{1+s_i}{1-s_i}\right]$$

and

$$l^-(s) = \frac{1}{2}\frac{1}{w}\sum_{i\in[w]}\left[\log\frac{1-s_i^2}{4} - s_i^*\log\frac{1+s_i}{1-s_i}\right].$$

Then

$$\overline{\mathcal{L}}(s,\bar{y}) = \begin{cases} l^+(s) & \text{if } \langle s_*, \log\frac{1+s}{1-s}\rangle \geq 0 \Leftrightarrow \bar{y} = \mathbb{1}_t = y_*, \\ l^-(s) & \text{otherwise, } (\Leftrightarrow \bar{y} = -y_*). \end{cases}$$

Observe, $l^+(s) \equiv l^-(-s)$ and vice versa; which suggests it is enough to maximize l^+. It also implies that the symmetric solution is the only other maximizer of the expected LLF. l^+ is concave and continuously differentiable on $(-1,+1)^w$. Therefore setting its gradient to 0 and solving for the skill satisfying this equality will yield a necessary condition on the maximizer:

$$\frac{\partial}{\partial s_i}l^+(s) = \frac{1}{2w}\left(\frac{1}{1-s_i^2}(-2s_i) + s_i^*\frac{1-s_i}{1+s_i}\frac{(1-s_i)+(1+s_i)}{(1-s_i)^2}\right) = \frac{1}{w}\frac{-s_i+s_i^*}{1-s_i^2} = 0,$$

that is, $\bar{s}_i = s_i^*$. We conclude that, as promised,

$$\arg\max_{s,y}\overline{\mathcal{L}}(s,y) = \{(s_*,y_*),(-s_*,-y_*)\}, \qquad \square$$

7.2 The Convergence of the Log-Likelihood to its Expectation

7.2.1 Concentration of the LLF at a fixed parameter

Proof of Lemma 8. We wish to bound

$$\mathbb{P}\left(|\mathcal{L}(s,y\,|\,T) - \overline{\mathcal{L}}(s,y)\,| \geq \varepsilon\right)$$

for $(s,y) \in [-1+\lambda, 1-\lambda]^w \times \{\pm 1\}^t$ parameters. From the definitions,

$$\overline{\mathcal{L}}(\theta) - \mathcal{L}(\theta\,|\,T) = \frac{1}{2}\frac{1}{wt}\sum_{i,j}(s_i^* - T_{ij})y_i v_i,$$

therefore, for any $\alpha > 0$,

$$\begin{aligned}
\mathbb{P}\left(\overline{\mathcal{L}}\left(\theta\right) - \mathcal{L}\left(\theta \,|\, T\right) \geq \varepsilon\right) &= \mathbb{P}\left(\alpha \frac{1}{2} \sum_{i,j} (s_i^* - T_{ij}) y_j \log \frac{1+s_i}{1-s_i} > \alpha w t \varepsilon\right) \\
&= \mathbb{P}\left(e^{\alpha\left(\frac{1}{2}\sum_{i,j}(s_i^* - T_{ij}) y_j \log \frac{1+s_i}{1-s_i} - wt\varepsilon\right)} > 1\right) \\
&\leq \mathbb{E}\left[e^{\alpha\left(\frac{1}{2}\sum_{ij}(s_i^* - T_{ij}) y_j \log \frac{1+s_i}{1-s_i} - wt\varepsilon\right)}\right] \\
&= \prod_{i,j} \mathbb{E}\left[e^{\frac{\alpha}{2}(s_i^* - T_{ij}) y_j \log \frac{1+s_i}{1-s_i}} e^{-\alpha\varepsilon}\right],
\end{aligned}$$

where we used Markov's Inequality and the independence of the $(T_{i,j})$ in the last two steps. We will bound the expectation using Lemma 3, the Kearns and Saul's inequality. Making the substitutions $t = -2av_i^*$ and $p = (1 - s_i^*)/2$ into (4.1), we get that for any $a \in \mathbb{R}$,

$$\begin{aligned}
\mathbb{E}\left[e^{-av_i^*(T_{ij} - s_i^*)}\right] &= \frac{1+s_i^*}{2} \exp\left(-2av_i^* \frac{1-s_i^*}{2}\right) + \frac{1-s_i^*}{2} \exp\left(2av_i^* \frac{1+s_i^*}{2}\right) \\
&\leq \exp\left(\frac{s_i^*}{4v_i^*}(-2av_i^*)^2\right) = \exp\left(a^2 s_i^* v_i^*\right).
\end{aligned}$$

We will use this with $a = \pm\alpha \frac{y_j v_i}{2v_i^*}$. The different signs bound the deviations in the two different directions, with both combined leading to the standard absolute value form. We work with just '+', noting that the other case goes the same way. Continuing from where we left off, we first apply the inequality we just proved term-wise, then simplify:

$$\begin{aligned}
\prod_{i,j} \mathbb{E}\left[e^{\alpha\left(\frac{1}{2}(s_i^* - T_{ij}) \log \frac{1+s_i}{1-s_i} - \varepsilon\right)}\right] &\leq \prod_{i,j} \exp\left(-\alpha\varepsilon + \alpha^2 \frac{y_j^2 v_i^2}{4(v_i^*)^2} s_i^* v_i^*\right) \\
&\leq \exp\left(-\alpha w t \varepsilon + \alpha^2 \sum_{ij} \frac{y_j^2 s_i^*}{4v_i^*} v_i^2\right).
\end{aligned}$$

Now we minimize the exponent over α. The exponent has the form $x\alpha^2 - y\alpha$, so setting the derivative to 0 we get $2x\alpha - y = 0$, meaning the optimum is $\alpha^* = \frac{y}{2x}$. At that point

the value of the exponent is $x\frac{y^2}{4x^2} - y\frac{y}{2x} = -\frac{y^2}{4x}$. Consequently, we get

$$\begin{aligned}
\mathbb{P}\left(\overline{\mathcal{L}}\left(\theta\right) - \mathcal{L}\left(\theta \,|\, T\right) \geq \varepsilon\right) &\leq \exp\left(-\frac{w^2 t^2 \varepsilon^2}{4\sum_{ij} \frac{y_j^2 s_i^*}{4v_i^*} v_i^2}\right) \\
&\leq \exp\left(-\frac{w^2 t \varepsilon^2}{\sum_i \frac{s_i^*}{v_i^*} v_i^2}\right) \\
&\leq \exp\left(-\frac{w^2 t \varepsilon^2}{\log^2(2\lambda^{-1} - 1)\sum_i \frac{s_i^*}{v_i^*}}\right),
\end{aligned}$$

where we first used $y_j^2 \leq 1$, then $v_i^2 \leq \log^2 \frac{2-\lambda}{\lambda} = \log^2(2\lambda^{-1} - 1)$, which are true due to the constraints on the domain. □

7.2.2 The covering number of our space

Proof of Lemma 9. We look for an upper bound on $N(\varepsilon/3, \Theta, d)$, where we recall d is the pseudometric defined as

$$d((s, y), (s', y')) \doteq \left\|\mathcal{L}\left(s, y \,|\, \cdot\right) - \mathcal{L}\left(s', y' \,|\, \cdot\right)\right\|_\infty$$

on the parameter space $\Theta = [-1 + \lambda, 1 - \lambda]^w \times \{\pm 1\}^t$. By construction, any ε-cover of (Θ, d) is an ε-cover of $(\mathcal{F} = \{\mathcal{L}\left(\theta \,|\, \cdot\right) : \theta \in \Theta\}, \|\cdot\|_\infty)$ and vice versa. Therefore we want to equivalently bound $N(\varepsilon/3, \mathcal{F}, \|\cdot\|_\infty))$. Here we abused notation slightly by using a norm in place of a pseudometric – simply consider the metric induced by the norm.

We use results concerning covering numbers collected in Section 4.4. Lemma 6 shows that we can calculate the covering number for $(\Theta, \|\cdot\|_\infty)$, then use a Lipschitz constant L associated to the LLF and the pseudometric spaces in consideration through

$$N(\varepsilon, \mathcal{F}, \|\cdot\|_\infty) \leq N(\varepsilon/L, \Theta, \|\cdot\|_\infty).$$

Lemma 7 upper bounds the right hand side. In order to find the Lipschitz constant it suffices to calculate the gradient of the LLF with respect to the parameters, as, by the mean-value theorem and Hölder's inequality,

$$\left\|\mathcal{L}\left(\theta \,|\, \cdot\right) - \mathcal{L}\left(\theta' \,|\, \cdot\right)\right\|_\infty \leq \left\{\sup_{\substack{\theta \in \Theta, \\ x \in \{\pm 1\}^{w \times t}}} \left\|\nabla_\theta \mathcal{L}\left(\theta \,|\, x\right)\right\|_1\right\} \left\|\theta - \theta'\right\|_\infty.$$

Since $\Theta = S_\lambda^w \times \{\pm 1\}^t$ with $\lambda > 0$, the skills in the argument of the supremum are bounded away from ± 1. For the gradient we calculate bounds on the partial derivatives:

$$\left|\frac{\partial}{\partial s_i}\mathcal{L}\left(s, y \,|\, x\right)\right| = \left|\frac{1}{2w} \cdot \frac{-2s_i}{1-s_i^2} + \frac{1}{2wt}\sum_{j\in[t]} x_{ij}y_j \frac{2}{1-s_i^2}\right| = \frac{1}{w}\frac{|s_i - \overline{x_{i,\cdot}y_\cdot}|}{1-s_i^2},$$

$$\left|\frac{\partial}{\partial y_j}\mathcal{L}\left(s, y \,|\, x\right)\right| = \left|0 + \frac{1}{2wt}\sum_{i\in[w]} x_{ij}\log\frac{1+s_i}{1-s_i}\right| = \frac{1}{2t}\left|\overline{x_{\cdot,j}\log\frac{1+s_\cdot}{1-s_\cdot}}\right|,$$

where $\bar{\cdot}$ denotes the average over the index not displayed. Therefore,

$$\sup_{\substack{\theta\in\Theta,\\ x\in\{\pm1\}^{w\times t}}}\left|\frac{\partial}{\partial s_i}\mathcal{L}\left(s, y \,|\, x\right)\right| = \frac{1}{w}\sup_{s\in S_\lambda}\frac{s+1}{1-s^2} = \frac{1}{w}\frac{1}{1-(1-\lambda)} = \frac{1}{\lambda w},$$

$$\sup_{\substack{\theta\in\Theta,\\ x\in\{\pm1\}^{w\times t}}}\left|\frac{\partial}{\partial y_j}\mathcal{L}\left(s, y \,|\, x\right)\right| = \frac{1}{2t}\sup_{s\in S_\lambda}\log\frac{1+s}{1-s} = \frac{1}{2t}\log\left(2\lambda^{-1}-1\right),$$

and hence $L = \frac{1}{\lambda} + \frac{1}{2}\log\left(2\lambda^{-1}-1\right)$ is a Lipschitz constant for the function $\theta \mapsto \mathcal{L}\left(\theta \,|\, \cdot\right)$ as a map between $(\Theta, \|\cdot\|_\infty)$ and $\mathcal{F}$. What remains is to calculate the ε-covering number of $(\Theta, \|\cdot\|_\infty)$. We note $\Theta \subseteq [-1,+1]^w \times [-1,+1]^t = [-1,+1]^{w+t}$, which is exactly the unit ball B in $\mathbb{R}^{w+t}$ when using $\|\cdot\|_\infty$, so we directly apply Lemma 7 bounding exactly this quantity. For any $\varepsilon > 0$

$$N(\varepsilon, \Theta, \|\cdot\|_\infty) \leq N(\varepsilon, B, \|\cdot\|_\infty) \leq \left(\frac{2}{\varepsilon}+1\right)^{w+t}.$$

Combining these, we see

$$N(\varepsilon/3, \mathcal{F}, \|\cdot\|_\infty) \leq N(\varepsilon/(3L), \Theta, \|\cdot\|_\infty) \leq \left(\frac{6L}{\varepsilon}+1\right)^{w+t}. \qquad \square$$

7.2.3 High probability uniform deviation bound

Proof of Lemma 10: Inverting the Concentration Bound. The goal is a bound of the form

$$\mathbb{P}\left(\sup_{s,y}\left|\mathcal{L}\left(s,y\right) - \overline{\mathcal{L}}\left(s,y\right)\right| < \varepsilon_{s_*}(\delta, w, t)\right) > 1-\delta$$

for every $\delta > 0$, given that

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |\mathcal{L}(\theta \,|\, T) - \overline{\mathcal{L}}(\theta)| \geq \varepsilon\right) \leq 2\left(\frac{6L}{\varepsilon} + 1\right)^{w+t} \exp\left(-\frac{w^2 t\varepsilon^2}{9\log^2(2\lambda^{-1}-1)\sum_i \frac{s_i^*}{v_i^*}}\right)$$

for any $\varepsilon > 0$ and parameter settings; here $L = \left\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\right\}$. We simply set the right hand side to be smaller than δ and rearrange the inequality with the goal of finding the smallest satisfying ε in terms of δ. We get the following sequence of equivalent inequalities:

$$2\left(\frac{6L}{\varepsilon} + 1\right)^{w+t} \exp\left(-\frac{2w^2 t\varepsilon^2}{9\log^2(2\lambda^{-1}-1)\sum_i \frac{s_i^*}{v_i^*}}\right) \leq \delta \qquad \Longleftrightarrow$$

$$(w+t)\log\left(\frac{6L}{\varepsilon} + 1\right) - \frac{2w^2 t\varepsilon^2}{9\log^2(2\lambda^{-1}-1)\sum_i \frac{s_i^*}{v_i^*}} \leq \log(\delta/2) \qquad \Longleftrightarrow$$

$$\left[(w+t)\log\left(\frac{6L}{\varepsilon} + 1\right) - \log(\delta/2)\right]\left(9\log^2(2\lambda^{-1}-1)\sum_i \frac{s_i^*}{v_i^*}\right) \leq 2w^2 t\varepsilon^2 \qquad \Longleftrightarrow$$

$$3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\frac{1}{2wt}\left[(w+t)\log\left(\frac{6L}{\varepsilon} + 1\right) + \log(2/\delta)\right]} \leq \varepsilon. \tag{7.1}$$

As opposed to directly solving for ε, we have arrived at an inequality -which we will write as $f(\varepsilon) \leq \varepsilon$ for short- with the variable we would like to bound appearing on both sides. To get a practical upper bound on ε, observe that f is decreasing in ε, therefore if some ε_* satisfies the inequality, so do larger values. In fact, let ε_* be such that $f(\varepsilon_*) = \varepsilon_*$, which must (uniquely) exist. Figure 7.1 illustrates these constructs visually. The plan is to select some $l \leq f(\varepsilon_*)$, then

$$l \leq f(\varepsilon_*) = \varepsilon_* \implies f(l) \geq f(\varepsilon_*) = \varepsilon_*\,,$$

so $\varepsilon \geq f(l)$ is sufficient for Equation (7.1) to hold. We set l to be the asymptote of f,

$$3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\frac{1}{2wt}\log(2/\delta)},$$

which clearly satisfies $l \leq f(\varepsilon)$ for any ε in fact. There we use $f(l)$,

$$3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\frac{1}{2wt}\left[(w+t)\log\left(\frac{6L}{l} + 1\right) + \log(2/\delta)\right]},$$

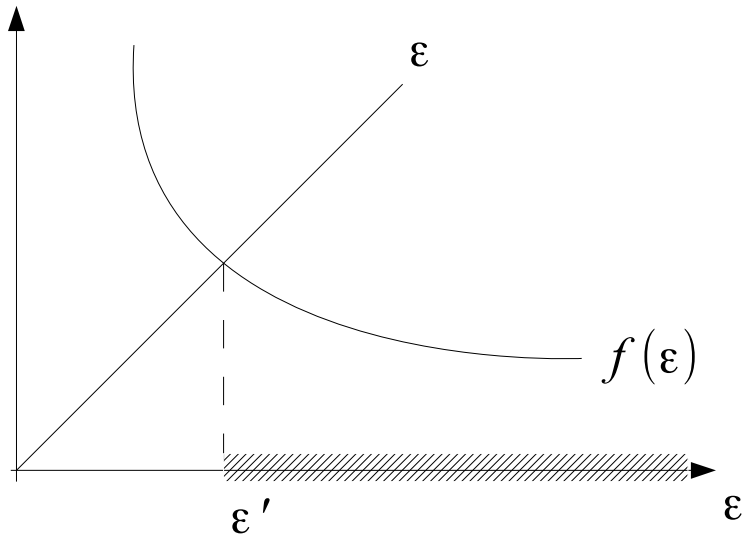

Figure 7.1: An illustration of the set of ε that satisfy Equation (7.1).
The shaded interval on the x-axis is the set of satisfying ε. We hope to find the smallest such ε, ε', though we only do so approximately.

as the $\varepsilon_{s_*}(\delta, w, t)$ we set out to find. Indeed, substituting L and l we arrive at the expression shown in the statement of the lemma. □

7.3 Bounds on the Error of the Weight Estimate

Proofs of Theorem 2 and Theorem 3. Both claims bound the ℓ_1 norm between the recovered weights and the true weights with the appropriate sign, given that $\overline{\mathcal{L}}(s_*, \mathbb{1}) - \overline{\mathcal{L}}(s, y) \leq \varepsilon$. That is, if $\operatorname{sgn}\left(\sum_j y_j\right) \geq 0$, we are closer to the global max of $y_* = \mathbb{1}$ and therefore measure the quality of the weights as the distance to the true weights v_*, but if $\operatorname{sgn}\left(\sum_j y_j\right) \leq 0$ we are near the negated peak and therefore recover $-v_*$. For the purposes of this analysis we WLOG assume we are in the former situation, that is, $\operatorname{sgn}\left(\sum_j y_j\right) \geq 0$.

The arguments are essentially the same for Theorems 2 and 3, except for a divergence towards the end. For convenience we pass to the weight domain, and interpret $\overline{\mathcal{L}}(\cdot, y)$ accordingly. Additionally, we note that y only interacts with the expected LLF (cf. Equation (5.2)) values through its average. Consequently, we will view $\overline{\mathcal{L}}$ as a $\overline{\mathcal{L}} : \mathbb{R}^w \times [-1, 1] \to \mathbb{R}$ map,

defined by

$$\overline{\mathcal{L}}\,(v,y) = \frac{1}{2}\frac{1}{w}\sum_{i=1}^{w}\left[\log\frac{1-\sigma(v_i)^2}{4} + y\sigma(v_i^*)v_i\right].$$

Recall that the $\sigma(\cdot)$ function transforms weights to skills; see Section 7.7 for its properties. Note that we may still use s_i in place of $\sigma(v_i)$ for a simpler notation. Figure 7.2 illustrates this function for one worker. The contours centred around the two maxima at the edges of the plot demonstrate the pre-images we set out to bound.

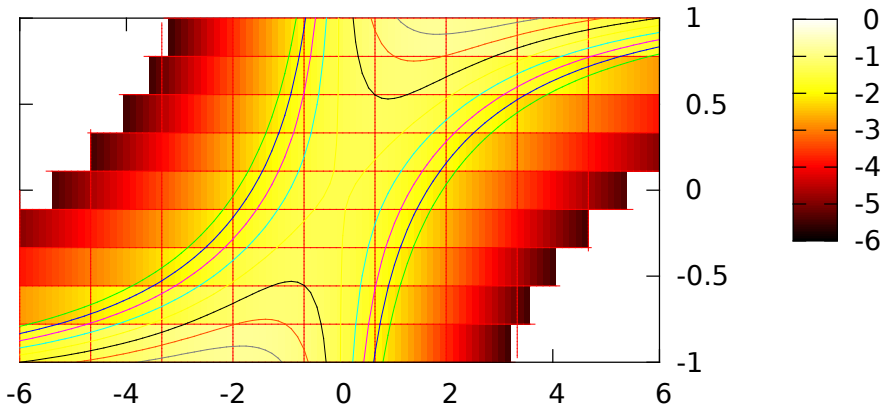

Figure 7.2: The contour plot of $\overline{\mathcal{L}}\,(v,y)$ for $w=1$ and continuous label $y\in[-1,1]$. Weights are shown on the horizontal axis, the continuous label on the vertical axis. For the generation of this plot the true skill was set to 2.2.

The goal is to upper bound

$$\max\{\|v-v_*\|_1 \;:\; \overline{\mathcal{L}}\,(v_*,1) - \overline{\mathcal{L}}\,(v,y) \le \varepsilon\}\,. \tag{7.2}$$

The difference can be calculated to be

$$\begin{aligned}\overline{\mathcal{L}}\,(v_*,1) - \overline{\mathcal{L}}\,(v,y) &= \frac{1}{2}\frac{1}{w}\sum_{i=1}^{w}\left[\log\frac{1-(s_i^*)^2}{4} - \log\frac{1-(s_i)^2}{4} + 1\cdot s_i^*v_i^* - ys_i^*v_i\right]\\ &= \frac{1}{2}\frac{1}{w}\sum_{i=1}^{w}\left[\log\frac{1-(s_i^*)^2}{1-(s_i)^2} + s_i^*(v_i^* - yv_i)\right].\end{aligned}$$

Observe that it is separable in i; we will work with just one term. Denote

$$A_{v_*}(v,y) \doteq \log\frac{1-s_*^2}{1-s^2} + s_*(v_* - yv)\,.$$

Equation (7.2) can be written as

$$\max\{\|v - v_*\|_1 \ : \ \frac{1}{w}\sum_{i=1}^{w} A_{v_i^*}(v, y) \le 2\varepsilon\}\,.$$

Noting that for any fixed v, $A_{v_i^*}(v, \cdot)$ is an increasing function on $0 \le y \le 1$, we see

$$\{v : A_{v_i^*}(v, y) \le c\} \subset \{v : A_{v_i^*}(v, 1) \le c\}$$

for any threshold $c > 0$. Therefore, Equation (7.2) is controlled by

$$\max\{\|v - v_*\|_1 \ : \ \frac{1}{w}\sum_{i=1}^{w} A_{v_i^*}(v, 1) \le 2\varepsilon\}\,. \tag{7.3}$$

For brevity, we start using $A_{v_i^*}(v)$ to denote $A_{v_i^*}(v, 1)$. In fact, let $l_{v_i^*} : \mathbb{R} \to \mathbb{R}$ be a lower bound for $A_{v_i^*}(\cdot, 1)$. Then Equation (7.3) is upper bounded by

$$\max\{\|v - v_*\|_1 \ : \ \frac{1}{w}\sum_{i=1}^{w} l_{v_i^*}(v) \le 2\varepsilon\}.$$

For the following steps we consider distributing the threshold 2ε amongst the terms.

$$\begin{aligned}
&\max\{\|v - v_*\|_1 \ : \ \frac{1}{w}\sum_{i=1}^{w} l_{v_i^*}(v) \le 2\varepsilon\} \\
&= \max\left\{\|v - v_*\|_1 \ : \ \forall i \in [w] \quad l_{v_i^*}(v) \le \varepsilon_i, \sum_{i=1}^{w}\varepsilon_i \le 2w\varepsilon, \varepsilon_i \ge 0\right\} \\
&= \max\left\{\sum_{i=1}^{w}\max\left\{|v_i - v_i^*| \ : \ l_{v_i^*}(v) \le \varepsilon_i\right\} \ : \ \sum_{i=1}^{w}\varepsilon_i \le 2w\varepsilon, \varepsilon_i \ge 0\right\}.
\end{aligned} \tag{7.4}$$

At this point the inner max can be optimized depending on the exact form of $l_{v_i^*}$. It is in the form of the lower-bound that Theorem 2 and Theorem 3 differ. However both rely on the same initial analysis of the A_{v_*} function, displayed on Figure 7.3. The idea is to lower bound it with a quadratic function fitted to it at its minimizer. The appropriate such function can be found by taking a second derivative:

$$\begin{aligned}
\frac{dA_{v_*}}{dv}(v) &= \sigma(v) - \sigma(v_*)\,, \\
\frac{d^2A_{v_*}}{d^2v}(v) &= \frac{1}{2}(1 - \sigma^2(v))\,.
\end{aligned}$$

There is some difficulty, as for extreme values of v the function A_{v_*} tends to a linear function. Nonetheless, we can argue locally. The first claim will lead to Theorem 2.

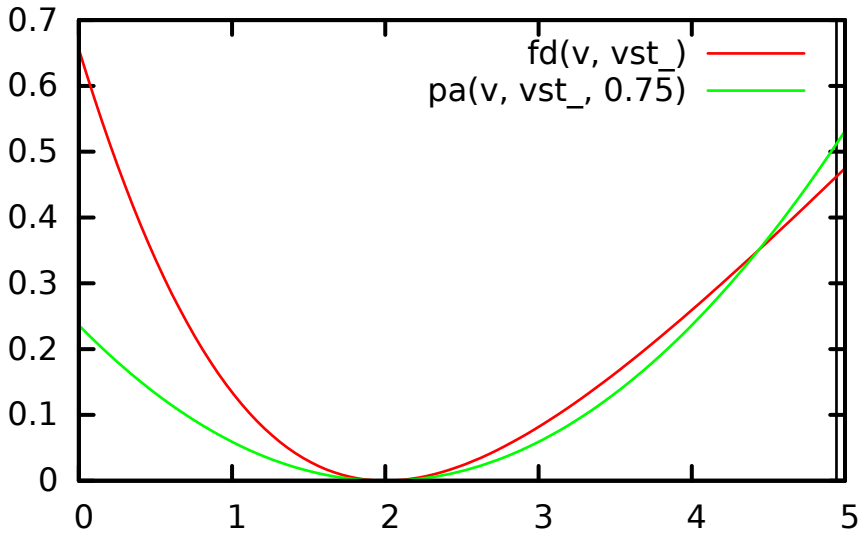

Figure 7.3: The plot of $A_2(v)$ in red and $l_q^{0.75}(v)$ in green.
The vertical line near $v = 5$ is v_λ for $\lambda = \sqrt{2/10^5}$.

Claim 1. *Let $A(v)$ be as defined above, for any fixed (and hidden) v_*. Then, for any $0 < \lambda < 1$ there is an open interval I around v_* such that for all $v \in I$*

$$A(v) \geq \frac{\gamma^2}{4}(1 - \sigma^2(v_*))(v_* - v)^2 \doteq l_q^\gamma(v).$$

Additionally, since A is convex, outside of this domain we can trivially lower bound it by the appropriate constant function.

Proof. By construction l_q^γ for $\gamma = 1$ and A have the same second derivative at v_*, with $l_q^{\gamma\prime\prime}(v_*) < A''(v_*)$ for any $\gamma < 1$. Moreover these two functions have the same value and (first) derivative at v_*. Since both functions are (at least) twice continuously differentiable, there is an open neighbourhood around v_* where $l_q^\gamma(v) \leq A(v)$. □

Corollary 3. *Fix any $0 < \gamma < 1$, v_* in the allowed domain. There exists $\varepsilon_0 > 0$ such that for any $\varepsilon \in (0, \varepsilon_0)$ we have*

$$A(v) \leq \varepsilon \implies l_q^\gamma(v) \leq A(v),$$

which, consequently, yields

$$|v - v_*| \leq \frac{2}{\gamma}\sqrt{\frac{\varepsilon}{1 - s_*^2}}.$$

Proof. Choose $\varepsilon = \min\{l_q^\gamma(I_l), l_q^\gamma(I_r)\}$, where (I_l, I_r) is the open interval from Claim 1. This guarantees the first implication. Next, we rearrange $l_q^\gamma = \frac{\gamma^2}{4}(1-s_*^2)(v_*-v)^2 \le \varepsilon$ to be

$$|v - v_*| \le \frac{2}{\gamma}\sqrt{\frac{\varepsilon}{1-s_*^2}}\,. \qquad \square$$

Under the assumptions of Corollary 3 we can continue from Equation (7.4):

$$\begin{aligned}\max\left\{\sum_{i=1}^{w}\max\left\{|v_i - v_i^*| \;:\; l_{v_i^*}(v) \le \varepsilon_i\right\} \;:\; \sum_{i=1}^{w}\varepsilon_i \le 2w\varepsilon, \varepsilon_i \ge 0\right\} \\ \le \max\left\{\sum_i \frac{2}{\gamma}\sqrt{\frac{\varepsilon_i}{1-(s_i^*)^2}} \;:\; \sum_i \varepsilon_i \le 2w\varepsilon, \varepsilon_i \ge 0\right\}.\end{aligned}$$

We will find this maximum now. For intuition, we make the transformation $x_i = \sqrt{\varepsilon_i}$, denote $\rho = \frac{1}{\sqrt{1-(s_i^*)^2}}$. Then the problem can be written as

$$\max_x \quad \frac{2}{\gamma}\langle x, \rho\rangle \;\text{ s.t. }\; \sum x_i^2 \le 2w\varepsilon, x_i \ge 0.$$

We know the inner product is maximized by aligned vectors with largest possible length, therefore the maximizing x (denoted simply by x here on) is $C\rho$ for some $C > 0$. We need to set its value so that it satisfies the equality constraint:

$$2w\varepsilon = \sum_i x_i^2 = \sum_i \frac{C^2}{1-(s_i^*)^2},$$

which implies $C = \sqrt{\frac{2w\varepsilon}{\sum_i \frac{1}{1-(s_i^*)^2}}}$. Therefore the maximal value is

$$\langle x, \rho\rangle = \sqrt{\frac{2w\varepsilon}{\sum_i \frac{1}{1-(s_i^*)^2}}}\sum_i \frac{1}{1-(s_i^*)^2} = \sqrt{2w\varepsilon\sum_i \frac{1}{1-(s_i^*)^2}}.$$

To summarize, we have shown that under the given conditions

$$\max\{\|v - v_*\|_1 \;:\; \overline{\mathcal{L}}(v_*, 1) - \overline{\mathcal{L}}(v, y) \le \varepsilon\} \le \frac{2}{\gamma}\sqrt{2w\varepsilon\sum_i \frac{1}{1-(s_i^*)^2}},$$

thus proving Theorem 2.

The merit of Theorem 3 is that it applies for any ε-difference in the function values. Towards this, we use a different strategy to lower bound A_{v_*}; Conjecture 1 gives us one. Note that the inequality postulated in this conjecture is in fact $A_{v_*}(v) \geq l_q^\gamma(v)$ with the choice $\gamma = \sqrt{\frac{2}{-\log \lambda}}$.

Corollary 4. *Assuming Conjecture 1 holds, for any* $0 < \lambda < 0.0067$, $v_* \in [-\infty, \infty]^w$, $v \in [-v_\lambda, v_\lambda]^w$ *and* $\varepsilon > 0$ *we have*

$$A_{v_*}(v) \leq \varepsilon \implies |v - v_*| \leq \sqrt{\frac{2\varepsilon \log(1/\lambda)}{1 - s_*^2}}\,.$$

Proof. By Conjecture 1 and our assumption, for any $0 < \lambda < 0.0067$, $v_* \in [-\infty, \infty]^w$, and any $v \in [-v_\lambda, v_\lambda]$

$$\frac{1}{2\log(1/\lambda)}(1 - s_*^2)(v_* - v) \leq A_{v_*}(v) \leq \varepsilon\,.$$

Consequently,

$$|v - v_*| \leq \sqrt{\frac{2\varepsilon \log(1/\lambda)}{1 - s_*^2}}.$$ □

What remains is to find the worst distribution of errors amongst the workers, as it was done in the case of Theorem 2. Under the assumptions of Corollary 4 we can continue from Equation (7.4):

$$\begin{aligned}
&\max\left\{\sum_{i=1}^w \max\left\{|v_i - v_i^*| \;:\; l_{v_i^*}(v) \leq \varepsilon_i\right\} \;:\; \sum_{i=1}^w \varepsilon_i \leq 2w\varepsilon, \varepsilon_i \geq 0\right\} \\
&\quad \leq \max\left\{\sum_i \sqrt{\frac{2\varepsilon \log(1/\lambda)}{1 - s_*^2}} \;:\; \sum_i \varepsilon_i \leq 2w\varepsilon, \varepsilon_i \geq 0\right\}
\end{aligned}$$

This is the same optimization problem as before, scaled. We therefore get

$$2\sqrt{w\varepsilon \log(1/\lambda) \sum_i \frac{1}{1 - (s_i^*)^2}},$$

as the maximum, which completes the proof of Theorem 3. □

7.4 From Approximate Weights to Labels

7.4.1 Optimal Weights Close to Erring

Proof of Lemma 11. By Chernoff's bounding method, for any $r > 0$,

$$\mathbb{I}\left\{v_*^\top T \le \varepsilon\right\} = \mathbb{I}\left\{r(\varepsilon - v_*^\top T) \ge 0\right\} = \mathbb{I}\left\{e^{r(\varepsilon - v_*^\top T)} \ge 1\right\} \le e^{r(\varepsilon - v_*^\top T)}\,.$$

Taking expectations,

$$\mathbb{P}\left(v_*^\top T \le \varepsilon\right) \le e^{\lambda\varepsilon}\mathbb{E}\left[e^{-rv_*^\top T}\right]\,.$$

Berend and Kontorovich (2014) calculate this expectation based on Kearns and Saul's inequality. See their equation (11), with $\theta_i = (T_i - s_i^*)/2$, $w_i = v_i^*$, to get

$$\mathbb{E}\left[e^{-rv_i^*(T_i - s_i^*)}\right] \le e^{r^2 s_i^* v_i^*}.$$

We continue bounding the probability of the event of our interest, using the independence of the $(T_i)_i$, and their inequality term-wise:

$$\begin{aligned}
\mathbb{P}\left(v_*^\top T \le \varepsilon\right) &\le e^{r\varepsilon}\mathbb{E}\left[e^{-rv_*^\top T}\right] = e^{r\varepsilon}\prod_i \mathbb{E}\left[e^{-rv_i^* T_i}\right] \\
&\le e^{r\varepsilon}\prod_i e^{s_i^* v_i^*(r^2 - r)} = e^{r\varepsilon + s_*^\top v_*(r^2 - r)} \\
&= \exp\left\{r s_*^\top v_*\left(r - \left(1 - \tfrac{\varepsilon}{s_*^\top v_*}\right)\right)\right\}\,.
\end{aligned}$$

Recalling that $s_*^\top v_*$ is the total committee potential, Φ, we observe that the first claim of the Lemma is proven. Next, assume

$$\varepsilon \le \Phi\,,$$

then the minimizer in r of the RHS above is $r^* = \frac{1}{2}(1 - \frac{\varepsilon}{\Phi})$ – found by taking a derivative and setting it to 0:

$$\frac{d}{dr}\left[r\Phi\left(r - \left(1 - \frac{\varepsilon}{\Phi}\right)\right)\right] = (2r - 1)\Phi + \varepsilon = 0\,,$$

and yields

$$\mathbb{P}\left(v_*^\top T \le \varepsilon\right) \le \exp\left\{-\frac{\Phi}{4}\left(1 - \frac{\varepsilon}{\Phi}\right)^2\right\}\,. \qquad \square$$

7.4.2 A Modular Mistake Bound

Proof of Theorem 4. The proof of the two parts of the theorem are quite similar; the difference is that for (5.10a) we relax $C_2 + \log(2/\delta)$ to be $C_2 \log(2/\delta)$, while towards (5.10b) we work with the exact expression.

The theorem basically calculates Equation (5.7) for a specific value of ε, defined implicitly so that our bounds on $\mathbb{P}\left(v_*^\top T \leq \varepsilon\right)$ and $\mathbb{P}\left(\|v_* - \hat{v}\|_1 > \varepsilon\right)$ are closely related. To be precise, given t_1 and t_2 expressions with

$$\mathbb{P}\left(v_*^\top T \leq \varepsilon\right) \leq e^{-t_1} \quad \text{and} \quad \mathbb{P}\left(\|v_* - \hat{v}\|_1 > \varepsilon\right) \leq e^{-t_2},$$

we will set

$$t_1 = \begin{cases} \sqrt{t_2}, & \text{for (5.10a);} \\ \sqrt{t_2 + C_2}, & \text{for (5.10b).} \end{cases}$$

At this point we concentrate on just one of the two results, give a complete proof of it, then turn our attention to the other.

Proof of Equation (5.10b) By the high probability weight approximation assumption of this Theorem, we have $\varepsilon = C_1 \sqrt[4]{C_2 + \log(1/\delta)}$, or equivalently, $\delta = \exp\left\{-[(\varepsilon/C_1)^4 - C_2]\right\}$, where δ is an upper bound on $\mathbb{P}\left(\|v_* - \hat{v}\|_1 > \varepsilon\right)$. Therefore our t_1, t_2 settings become

$$t_1 = \frac{\Phi}{4}\left(1 - \frac{\varepsilon}{\Phi}\right)^2 \quad \text{and} \quad t_2 = (\varepsilon/C_1)^4 - C_2,$$

with t_1 given by Lemma 11, under the assumption that $\varepsilon \leq \Phi$. We solve $\sqrt{t_2 + C_2} = t_1$ for ε; the following equations are all equivalent given that $\varepsilon \leq \Phi$:

$$\begin{aligned} (\varepsilon/C_1)^2 = \sqrt{t_2 + C_2} = t_1 &= \tfrac{\Phi}{4}\left(1 - \tfrac{\varepsilon}{\Phi}\right)^2, \\ \varepsilon/C_1 &= \tfrac{\sqrt{\Phi}}{2}\left(1 - \tfrac{\varepsilon}{\Phi}\right), \\ \varepsilon &= C_1\sqrt{\Phi}\left(2 + \tfrac{C_1}{\sqrt{\Phi}}\right)^{-1}. \end{aligned}$$

We remark that

$$\varepsilon = C_1\sqrt{\Phi}\left(2 + \tfrac{C_1}{\sqrt{\Phi}}\right)^{-1} = \Phi\tfrac{C_1}{2\sqrt{\Phi} + C_1} \leq \Phi,$$

as required. Next we find the corresponding t_1.

$$1 - \frac{\varepsilon}{\Phi} = 1 - \frac{C_1\sqrt{\Phi}}{\Phi\left(2 + \frac{C_1}{\sqrt{\Phi}}\right)} = 1 - \frac{C_1}{\Phi\left(2\sqrt{\Phi} + C_1\right)}$$
$$= \frac{2\sqrt{\Phi}}{2\sqrt{\Phi} + C_1} = \left(1 + \frac{C_1}{2\sqrt{\Phi}}\right)^{-1},$$

therefore

$$t_1 = \frac{\Phi}{4}\left(1 - \frac{\varepsilon}{\Phi}\right)^2 = \frac{\Phi}{4}\left(1 + \frac{C_1}{2\sqrt{\Phi}}\right)^{-2}.$$

We trivially bound the probability of making an error:

$$\mathbb{P}\left(v_*^\top T \le \varepsilon\right) \le e^{-t_1} + e^{-t_2} = e^{-t_1} + e^{-(t_1^2 + C_2)} \le 2\exp\{-\min(t_1, t_1^2 + C_2)\}.$$

Proof of Equation (5.10a) As this proof is almost identical we skip some details. We start by noting that $C_2 + \log(1/\delta) \le C_2 \log(1/\delta)$ for $C_2, \log(1/\delta) \ge 1$. Then

$$\varepsilon = C_1[C_2 \log(1/\delta)]^{1/4} \quad \Leftrightarrow \quad \delta = \exp\{-(\tfrac{\varepsilon}{C_1\sqrt[4]{C_2}})^4\},$$

with $t_2 = (\frac{\varepsilon}{C_1\sqrt[4]{C_2}})^4$. For this case we set $t_1 = \sqrt{t_2}$ and solve it for ε. Using $C_3 \doteq C_1\sqrt[4]{C_2}$ in place of C_1 in the calculations of the previous case, we see

$$\varepsilon = C_3\sqrt{\Phi}\left(2 + \tfrac{C_3}{\sqrt{\Phi}}\right)^{-1},$$

and the corresponding t_1 is

$$\frac{\Phi}{4}\left(1 + \frac{C_3}{2\sqrt{\Phi}}\right)^{-2}.$$

Here

$$\mathbb{P}\left(v_*^\top T \le \varepsilon\right) \le e^{-t_1} + e^{-t_2} = e^{-t_1} + e^{-(t_1^2)} \le 2\exp\{-\min(t_1, t_1^2)\}. \qquad \square$$

7.5 Mistake Bounds

Proof of Theorem 5. Fix any $0 < \lambda < 0.0067$, $s_* \in S_\lambda^w$. Without loss of generality, as before, we assume that the true labels are all $+1$. Let $\hat{s}$ be the skill estimates returned by

the MLE over $\Theta = S_\lambda^w \times \{\pm 1\}^t$ and let $\hat{y} \in \{\pm 1\}^t$ be the corresponding label estimates (also returned by MLE). Let $\hat{v} = v(\hat{s})$. Under our assumptions, Theorem 3 gives a bound on $\|v(\hat{s}) - \chi v(s_*)\|_1$ in terms of $\varepsilon = \overline{\mathcal{L}}(s_*, \mathbb{1}) - \overline{\mathcal{L}}(\hat{s}, \hat{y})$, where $\chi = \operatorname{sgn}\left(\sum_j \hat{y}_j\right)$:

$$\|\hat{v} - \chi v(s_*)\|_1 \leq 2\sqrt{w\varepsilon \log(\tfrac{1}{\lambda}) \sum_i \frac{1}{1-(s_i^*)^2}}\,, \tag{7.5}$$

By (5.5), specifically since $s_* \in S_\lambda^w$, the value of ε is upper bound as

$$\varepsilon \leq 2 \sup_{\theta \in \Theta} \left| \mathcal{L}(\theta) - \overline{\mathcal{L}}(\theta) \right|. \tag{7.6}$$

This, by Lemma 10 is upper bounded by $2\varepsilon_{s_*}(\delta, w, t)$ with probability $1-\delta$, where

$$\varepsilon_{s_*}(\delta, w, t) \doteq 3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\frac{1}{2wt}\Big[(w+t)\log C + \log(2/\delta)\Big]}\,, \tag{7.7}$$

with C given by Equation (5.4). Combining (7.5),(7.6) with (7.7) shows that the condition of Theorem 4, namely that with probability $1-\delta$,

$$\|\hat{v} - v_*\|_1 \leq C_1 \sqrt[4]{C_2 + \log(1/\delta)}$$

holds provided we choose C_1, C_2 as

$$\begin{aligned}
C_1 &= 2\sqrt{6w\log(\lambda^{-1})\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\frac{1}{2wt}}\left(\sum_i \frac{1}{1-(s_i^*)^2}\right)} \\
&\leq 2\sqrt{w}\log(2/\lambda)\sqrt{6\sqrt{\frac{w}{2t}}\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\left(\frac{1}{w}\sum_i \frac{1}{1-(s_i^*)^2}\right)}\,, \\
C_2 &= (w+t)\log C + \log(2) \\
&= (w+t)\log\left(\frac{6\sqrt{2wt}\{\frac{1}{\lambda} + \frac{1}{2}\log(2\lambda^{-1}-1)\}}{3\log(2\lambda^{-1}-1)\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}}\sqrt{\log(2/\delta)}} + 1\right) + \log(2)\,.
\end{aligned}$$

Therefore, by Theorem 4, more specifically, by (5.10b), the probability of mislabelling a task when using $\operatorname{sgn}\left(\hat{v}^\top \cdot\right)$ on the observed labels of any given task, is upper bounded by

$$2\exp\left\{-\min\left(\tfrac{\Phi}{4}(1+r)^{-2},\ (\tfrac{\Phi}{4}(1+r)^{-2})^2 + C_2\right)\right\}.$$

where

$$r \doteq \frac{C_1}{2\sqrt{\Phi}} = \log(2/\lambda)\sqrt{6\sqrt{\frac{w}{2t}}\sqrt{\frac{1}{w}\sum_i \frac{s_i^*}{v_i^*}\left(\frac{1}{w}\sum_i \frac{1}{1-(s_i^*)^2}\right)}\,\overline{\Phi}^{-1}}\,. \qquad \square$$

7.6 Inference with Deterministic Weights

Proof of Lemma 12. Assume as before that $y_* = 1$. We bound $e(v) = \mathbb{P}\left(\operatorname{sgn}\left(v^\top T\right) \neq 1\right)$, by the same combination of Chernoff's bounding method and Kearns and Saul's inequality (Lemma 3) we have seen before. For any $\lambda \geq 0$,

$$\mathbb{I}\left\{\operatorname{sgn}\left(v^\top T\right) \neq 1\right\} \leq \mathbb{I}\left\{\exp(-\lambda v^\top T) \geq 1\right\} \leq e^{-\lambda v^\top T} = e^{-\lambda v^\top s_*}\, e^{-\lambda v^\top (T - s_*)}\,,$$

hence, by the independence of $(T_i)_i$,

$$e(v) \leq e^{-\lambda v^\top s_*} \prod_{i=1}^{w} \mathbb{E}\left[e^{-\lambda v_i (T_i - s_i^*)}\right]\,.$$

The Kearns-Saul bound gives

$$\mathbb{E}\left[e^{-\lambda v_i (T_i - s_i^*)}\right] \leq e^{\lambda^2 \frac{s_i^* v_i^2}{v_i^*}}\,,$$

Hence, defining $\rho(v) = \sum_{i=1}^{w} \frac{s_i^* v_i^2}{v_i^*}$,

$$e(v) \leq e^{-\lambda v^\top s_* + \lambda^2 \rho(v)}\,.$$

The minimizer of the exponent is $\lambda^* = \frac{v^\top s_*}{2\rho(v)}$, and with this choice we get

$$e(v) \leq e^{-\frac{1}{4}\frac{(v^\top s_*)^2}{\rho(v)}}\,. \qquad \square$$

7.7 Supplementary Results

Lemma 13. *Let $f, g : X \to \mathbb{R}$, with $a \in \arg\max_x f(x)$ and $b \in \arg\max_x g(x)$. Then*

$$f(a) - f(b) \leq 2 \sup_x |f(x) - g(x)|.$$

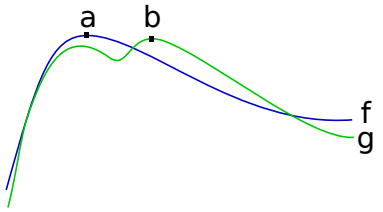

Figure 7.4: Functions of bounded uniform deviation have similar maxima.

Proof.

$$\begin{aligned} f(a) - f(b) &= f(a) - g(a) + g(a) - f(b) \\ &\leq f(a) - g(a) + g(b) - f(b) \\ &\leq 2 \sup_x |f(x) - g(x)| \end{aligned}$$

□

Fact 2 (The weight and sigmoid functions)**.** *We define the sigmoid function, and find its inverse to be*

$$\sigma(v) = \frac{1 - e^{-v}}{1 + e^{-v}}, \quad v = \log \frac{1 + \sigma(v)}{1 - \sigma(v)}.$$

We see that the latter is in fact the weight function. The following will be useful:

$$1 - \sigma(v) = \frac{2e^{-v}}{1 + e^{-v}}, \tag{7.8a}$$

$$1 + \sigma(v) = \frac{2}{1 + e^{-v}}. \tag{7.8b}$$

The first and second derivatives are

$$\frac{d}{dv}\sigma(v) = \frac{1}{2}(1 - \sigma^2(v)),$$

$$\frac{d^2}{d^2v}\sigma(v) = -\frac{1}{2}\sigma(v)(1 - \sigma^2(v)).$$

The following derivatives are also used:

$$\frac{d}{dv}\log(1 - \sigma^2(v)) = -\sigma(v),$$

$$\frac{d^2}{d^2v}\log(1 - \sigma^2(v)) = -\frac{1}{2}(1 - \sigma^2(v)).$$

Next, let us restate the weight function with our usual notation:

$$v(s) = \log \frac{1+s}{1-s}\,.$$

Its derivative is

$$\frac{d}{ds} v(s) = \frac{2}{1-s^2}.$$

Fact 3 (A lower bound)**.** *Let $s \in [-1,+1]$ and $v = v(s)$ be its corresponding weight. Then*

$$\frac{1}{2+sv} \leq \frac{s}{v}\,,$$

moreover, for $s \in [-1,+1]^w$ with corresponding weight vector v

$$\frac{1}{w} \sum_{i\in[w]} \frac{s_i}{v_i} \geq \frac{1}{2+\overline{\Phi}}\,,$$

where we recall $\overline{\Phi}$ is the average committee potential, $\frac{1}{w}\sum_{i\in[w]} s_i v_i$.

Proof of Fact 3. We first rearrange the inequality. For $v \leq 0$ and $v \geq 0$ we get

$$0 \leq v - \frac{2s}{1-s^2} \quad \text{and} \quad 0 \leq \frac{2s}{1-s^2} - v$$

respectively. We treat only the second case, the first is analogous. For the rest of the argument we write out the dependency of v on s. Note that at $s = 0$ ($v(s) = 0$) the right hand side evaluates to 0. Next, consider its derivative:

$$\frac{d}{ds}\left[\frac{2s}{1-s^2} - v\right] = \frac{2(s^2+1)}{(1-s^2)^2} - \frac{2}{1-s^2} = 2\frac{1-s^4}{(1-s^2)^2}\,.$$

We see the difference is increasing when $1 - s^4 \geq 0$; which is true for any s in the domain. We conclude that $0 \leq \frac{2s}{1-s^2} - v(s)$ holds for all $s \geq 0$ (equivalently, $v \geq 0$) and consider the first statement proven.

For the second statement we apply the first inequality term-wise, then use Jensen's inequality applied to the convex $\frac{1}{2+x}$:

$$\frac{1}{w}\sum_{i\in[w]} \frac{s_i}{v_i} \geq \frac{1}{w}\sum_{i\in[w]} \frac{1}{2+sv} \geq \frac{1}{2+\frac{1}{w}\sum_{i\in[w]} s_i v_i}\,. \qquad \square$$

References

Abbasi-Yadkori, Y., Bartlett, P. L., Chen, X., and Malek, A. (2015). Large-Scale Markov Decision Problems with KL Control Cost and its Application to Crowdsourcing. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1053–1062.

Amazon (2015). Introduction to Amazon Mechanical Turk. http://docs.aws.amazon.com/AWSMechTurk/2008-08-02/AWSMechanicalTurkRequester/IntroductionArticle.html.

Bartlett, P. (2013). Theoretical Statistics. Lecture 12. http://www.stat.berkeley.edu/~bartlett/courses/2013spring-stat210b/notes/12notes.pdf.

Berend, D. and Kontorovich, A. (2013). On the concentration of the missing mass. *Electron. Commun. Probab.*, 18:no. 3, 1–7.

Berend, D. and Kontorovich, A. (2014). Consistency of weighted majority votes. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3446–3454.

Bousquet, O., Boucheron, S., and Lugosi, G. (2004). Introduction to Statistical Learning Theory. In Bousquet, O., Luxburg, U. v., and Rätsch, G., editors, *Advanced Lectures on Machine Learning*, number 3176 in Lecture Notes in Computer Science, pages 169–207. Springer Berlin Heidelberg.

Bragg, J., Kolobov, A., Mausam, and Weld, D. S. (2014). Parallel Task Routing for Crowdsourcing. In *Proceedings of the Seconf AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2014, November 2-4, 2014, Pittsburgh, Pennsylvania, USA*.

Chen, E. (2012). Making the Most of Mechanical Turk: Tips and Best Practices. http://blog.echen.me/2012/04/25/making-the-most-of-mechanical-turk-tips-and-best-practices/.

Dalvi, N., Dasgupta, A., Kumar, R., and Rastogi, V. (2013). Aggregating Crowdsourced Binary Ratings. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 285–294, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Dawid, P., Skene, A. M., Dawid, A. P., and Skene, A. M. (1979). Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, pages 20–28.

Gao, C. and Zhou, D. (2013). Minimax Optimal Convergence Rates for Estimating Ground Truth from Crowdsourced Labels. *arXiv:1310.5764 [math, stat]*. version 5.

Ghosh, A., Kale, S., and McAfee, P. (2011). Who Moderates the Moderators?: Crowdsourcing Abuse Detection in User-generated Content. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, EC '11, pages 167–176, New York, NY, USA. ACM.

Ho, C.-J., Jabbari, S., and Vaughan, J. W. (2013). Adaptive Task Assignment for Crowdsourced Classification. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 534–542.

Ipeirotis, P. (2010). Be a Top Mechanical Turk Worker: You Need $5 and 5 Minutes | A Computer Scientist in a Business School. http://www.behind-the-enemy-lines.com/2010/10/be-top-mechanical-turk-worker-you-need.html.

Jaffe, A., Nadler, B., and Kluger, Y. (2015). Estimating the accuracies of multiple classifiers without labeled data. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015*.

Jung, H. J. (2014). Quality Assurance in Crowdsourcing via Matrix Factorization Based Task Routing. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, WWW Companion '14, pages 3–8, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Karger, D., Oh, S., and Shah, D. (2011a). Budget-optimal crowdsourcing using low-rank matrix approximations. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 284–291.

Karger, D. R., Oh, S., and Shah, D. (2011b). Iterative Learning for Reliable Crowdsourcing Systems. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.*, pages 1953–1961.

Karger, D. R., Oh, S., and Shah, D. (2013). Efficient crowdsourcing for multi-class labeling. In *ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems, SIGMETRICS '13, Pittsburgh, PA, USA, June 17-21, 2013*, pages 81–92.

Karger, D. R., Oh, S., and Shah, D. (2014). Budget-Optimal Task Allocation for Reliable Crowdsourcing Systems. *Operations Research*, 62(1):1–24.

Kearns, M. J. and Saul, L. K. (1998). Large Deviation Methods for Approximate Probabilistic Inference. In *UAI '98: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, University of Wisconsin Business School, Madison, Wisconsin, USA, July 24-26, 1998*, pages 311–319.

Kolobov, A., Mausam, and Weld, D. S. (2013). (JoCR) Joint Crowdsourcing of Multiple Tasks. In *First AAAI Conference on Human Computation and Crowdsourcing*.

Kumar, N. (2014). Effective Use of Amazon Mechanical Turk (MTurk). http://neerajkumar.org/writings/mturk/.

Li, H., Yu, B., and Zhou, D. (2013). Error Rate Bounds in Crowdsourcing Models. *arXiv:1307.2674 [cs, stat]*. arXiv: 1307.2674.

Liu, Q., Ihler, A. T., and Fisher III, J. (2015). Boosting Crowdsourcing With Expert Labels: Local vs. Global Effects. In *Information Fusion (Fusion), 2015 18th International Conference on*.

Liu, Q., Peng, J., and Ihler, A. T. (2012). Variational Inference for Crowdsourcing. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 701–709.

Nitzan, S. and Paroush, J. (1982). Optimal decision rules in uncertain dichotomous choice situations. *International Economic Review*, 23(2):289–297.

Pollard, D. (1984). *Convergence of Stochastic Processes*. Springer Verlag, New York.

Raginsky, M. and Sason, I. (2013). Concentration of Measure Inequalities in Information Theory, Communications, and Coding. *Foundations and Trends in Communications and Information Theory*, 10(1-2):1–247.

redditer (2014). New to Mturk, Here's what you should know. * /r/mturk. http://www.reddit.com/r/mturk/comments/1z4sma/new_to_mturk_heres_what_you_should_know/.

Welinder, P., Branson, S., Belongie, S., and Perona, P. (2010). The Multidimensional Wisdom of Crowds. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada.*, pages 2424–2432.

Whitehill, J., Ruvolo, P., Wu, T., Bergsma, J., and Movellan, J. R. (2009). Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 2035–2043.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. I. (2014). Spectral Methods meet EM: A Provably Optimal Algorithm for Crowdsourcing. *arXiv:1406.3824 [stat]*. arXiv: 1406.3824.

Zooniverse (2015). Snapshot Serengeti. http://www.snapshotserengeti.org/?utm_source=Zooniverse%20Home&utm_medium=Web&utm_campaign=Homepage%20Catalogue.

Appendix A

Supporting Material for Conjecture 1

Recall that Conjecture 1 stated the following:

For any $0 < \lambda < 0.0067$, $s_* \in [-1, 1]$, $s \in S_\lambda$ *we have*

$$\log \frac{1 - s_*^2}{1 - s^2} + s_*(v_* - v) \geq \frac{1 - s_*^2}{2 \log(1/\lambda)} (v_* - v)^2 \,. \tag{A.1}$$

Here, we will explain how this was derived and we present details of an argument that have some gaps in it but which may eventually lead to a proof of this conjecture.

First, recall that the left-hand-side of (A.1) was earlier called A_{v_*}: $A_{v_*}(v) = \log \frac{1-s_*^2}{1-s^2} + s_*(v_* - v)$ where we use the earlier adapted convention that v stands for $v(s)$, with $v(s) = \log \frac{1+s}{1-s}$ (and $v_* = v(s_*)$). Also recall that the right-hand-side of (A.1) is equal to $l_q^\gamma(v) = \frac{\gamma^2}{4}(1 - s_*^2)(v_* - v)^2$ with $\gamma = \sqrt{2/\log(1/\lambda)}$.

The conjecture is derived from the desire to lower bound A_{v_*} over $[-v_\lambda, v_\lambda]$ with a quadratic function of simple form that does not depend on v_*, where $v_\lambda = v(1 - \lambda)$ (note that $-v_\lambda = v(-1 + \lambda)$). For the sake of the argument, assume that $v_* \geq 0$; the argument for $v_* \leq 0$ is analogous.

The idea is that A_{v_*} is convex (this can be verified). Both A_{v_*} and l_q^γ are nonnegative, they are both zero at $v = v_*$ and thus their first derivatives are both zero at the same point. Noting that v_λ is an upper bound on v_* by assumption, we first consider the interval $[v_*, v_\lambda]$; the interval $[-v_\lambda, v_*]$ will be considered later.

If l_q^γ stays below A_{v_*} in a small neighborhood of v_* then, due to their convexity, if they first meet at $v = v_\lambda$ or later, l_q^γ will be below A_{v_*} on the whole interval $[v_*, v_\lambda]$. Solving $l_q^\gamma(v_\lambda) = A_{v_*}(v_\lambda)$ for γ, under the assumption that $0 < \lambda < 1/2$, we get that if

$$\gamma \leq f_v(v_*, \lambda) \doteq \frac{2}{v_\lambda - v_*} \sqrt{\frac{1}{1-(s_*)^2} \left(\log \frac{1-(s_*)^2}{1-(1-\lambda)^2} - s_*(v_\lambda - v_*) \right)}$$

then

$$l_q^\gamma(v_\lambda) \leq A(v_\lambda)\,. \tag{A.2}$$

Now, we need $\gamma \leq f_v(v_*, \lambda)$ to hold regardless the value of v_*. Our approach will be to use some $\gamma(\lambda) \leq \min_{-v_\lambda \leq v_* \leq v_\lambda} f_v(v_*, \lambda)$. Since f is non-negative and we take the minimum over a bounded domain, a minimizer actually exists. Let us denote it by $\check{v}$. We conjecture that for $v_* \geq 0$ the minimizer is in $[0, v_\lambda]$; supporting material appears in Appendix A.1. In fact, we notice that $\check{v} \approx \log(v_\lambda) \approx \log(-\log(\lambda))$. See Figure A.1 for a visualization. Clearly, the latter will result in a slightly higher value, so we will have to correct for this. The function f_v appears convex in v, so will use the tangent at the place of the approximation as a lower bound. This requires us to upper bound the (unsigned) slope of the function f_v at the approximation, and also find how far the approximate and true minimizers are. But let us study $f_v(\cdot, \lambda)$ at its approximate minimum first. It is more natural to view λ as e^{-p} for this argument. Then $v_\lambda = \log \frac{1+(1-e^{-p})}{1-(1-e^{-p})} = \log(2e^p - 1)$, which is asymptotically the same as p. We use $\hat{v} = \log p$ as the approximate minimum for the rest of the argument. Next, we collect the precise or approximate values of each term in $f_v(\hat{v}, \lambda)$:

$$\begin{aligned}
s_\lambda &= 1 - e^{-p}\,, \\
v_\lambda &= \log(2e^p - 1) \approx p\,, \\
v_\lambda - \hat{v} &\approx p - \log p\,, \\
\hat{s} &= \frac{1 - e^{-\hat{v}}}{1 + e^{-\hat{v}}} = \frac{1 - e^{-\log p}}{1 + e^{-\log p}} = \frac{p-1}{p+1}\,, \\
\frac{1}{1-(\hat{s})^2} &= \left(1 - \left[\frac{1-p}{1+p} \right]^2 \right)^{-1} = \frac{(1+p)^2}{4p}\,, \\
\log \frac{1-(\hat{s})^2}{1-(1-e^{-p})^2} &= \log \left(\frac{4p}{(1+p)^2} \cdot \frac{e^p}{2 - e^{-p}} \right) \\
&= p - \log(2 - e^{-p}) + \log 4 + \log p - 2\log(1+p) \\
&\approx p - \log p\,.
\end{aligned}$$

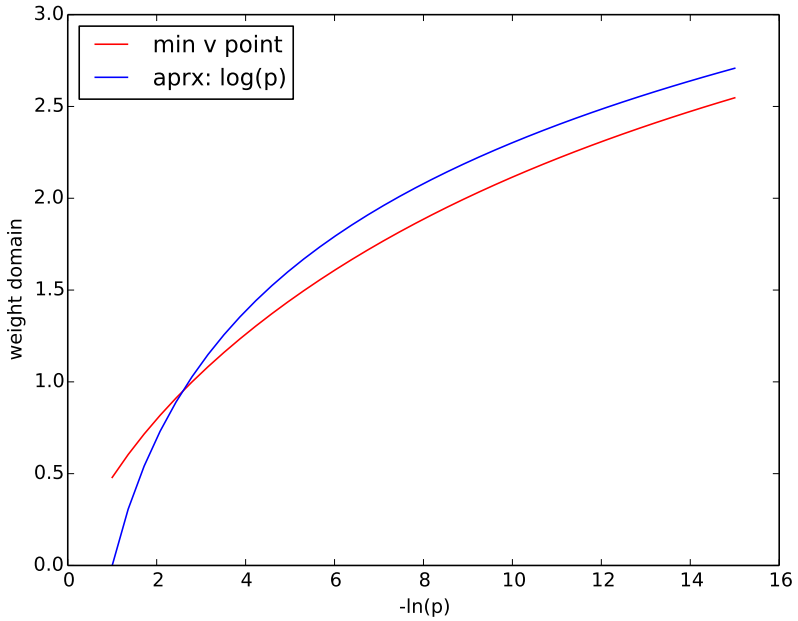

Figure A.1: Comparing the true and approximate minimizers of $f_v(\cdot, \lambda)$.
We compare, as a function of $p = -\log(\lambda)$, the location of the true minimizer of f_v in v, calculated numerically, and the approximate location used, $\log(p)$. We see that they behave similarly, with the approximate appearing to be consistently larger after some point. *Note:* label on the x-axis is wrong, it should be p.

After combining these and executing a number of other approximation we get

$$\inf_v f_v(v, \lambda) \approx f_v(\log p, \lambda) \approx \sqrt{\frac{2(p+1)}{p(p - \log p)}} \approx \sqrt{\frac{2}{p}}\,. \tag{A.3}$$

In fact, one can show $f_v(\log p, \lambda) \asymp \sqrt{\frac{2}{p}}$.

Next, we find the derivative of $f_v(\cdot, \lambda)$:

$$\frac{\partial f}{\partial v} f(v, \lambda) = \frac{f(v, \lambda)}{2(v_\lambda - v)} \left[s_v(v_\lambda - v) - 2(f^{-2}(v, \lambda) - 1) \right]. \tag{A.4}$$

We are particularly interested in its value at $\hat{v}$, the approximate minimum we consider.

Substituting the approximate value of $f_v(\log p, \lambda)$ into the partial derivative we get

$$\begin{aligned}\frac{\partial f}{\partial v} f(v,\lambda)|_{v=\log p} &\approx \sqrt{\frac{2(p+1)}{p(p-\log p)}} \frac{1}{p-\log p}\left[\frac{1}{2}\frac{p-1}{p+1}(p-\log p) - \frac{p(p-\log p)}{2(p+1)} + 1\right] \\ &= \sqrt{\frac{2(p+1)}{p(p-\log p)}}\left[1 - \frac{1}{2(p+1)}\right], \end{aligned} \tag{A.5}$$

which, again, behaves like $\sqrt{\frac{2}{p}}$ asymptotically. In order to guarantee our estimate is an upper bound on absolute value of the derivative we can choose to use, say $p^{-1/4}/50$.

At this point we visualize these bounds, but fail to continue the argument: we have not been able to upper bound the distance of the true and approximate minimizers, and have not verified that $f_v(\cdot, \lambda)$ is indeed convex for every λ under consideration. Nonetheless, Figure A.2 illustrates that both simply $\sqrt{2/p}$ and the expression derived from our analysis indeed lower bound and match $\min_v f_v(v, \lambda)$ well. This is in fact what led to the choice of $\gamma = \sqrt{\frac{2}{-\log \lambda}}$ in Conjecture 1. Let us now continue filling the remaining gaps.

A.1 Range restriction on $\check{v}$ and the inequality over $[-v_\lambda, v_*]$

We now motivate making the conjecture that for $v_* \geq 0$ we have $\check{v} \in [0, v_\lambda]$. That is, it is enough to choose $\gamma(\lambda) \leq \min_{0 \leq v_* \leq v_\lambda} f_v(v_*, \lambda)$ as it will be equal to $\min_{-v_\lambda \leq v_* \leq v_\lambda} f_v(v_*, \lambda)$. The function A_{v_*} is symmetric for $v_* = 0$; as v_* increases towards v_λ it continues to be steep on $(-\infty, v_*]$ while it flattens out on $[v_*, \infty)$. Since the lower bound is symmetric in v_*, this suggests the conjecture will hold. The only concern is that during this process $|-v_\lambda - v_*|$ grows while $|v_\lambda - v_*|$ decreases. We find that this does not become an issue.

We provide a sequence of images to illustrate that as v_* moves towards v_λ from 0 the quadratic lower bound used on $[v_*, v_\lambda]$ clearly continues to hold on $[-v_\lambda, v_*]$. To generate this sequence of images we set $\lambda = e^{-6}$, $\gamma = \sqrt{2/6}$ accordingly, and plotted A_{v_*} and l_q^λ for $v_* \in [6]$. Figure A.3 and Figure A.4 depict the first 6 of these, Figure A.5 may be used to verify on a smaller scale that the quadratic function is still indeed lower bounding the required function.

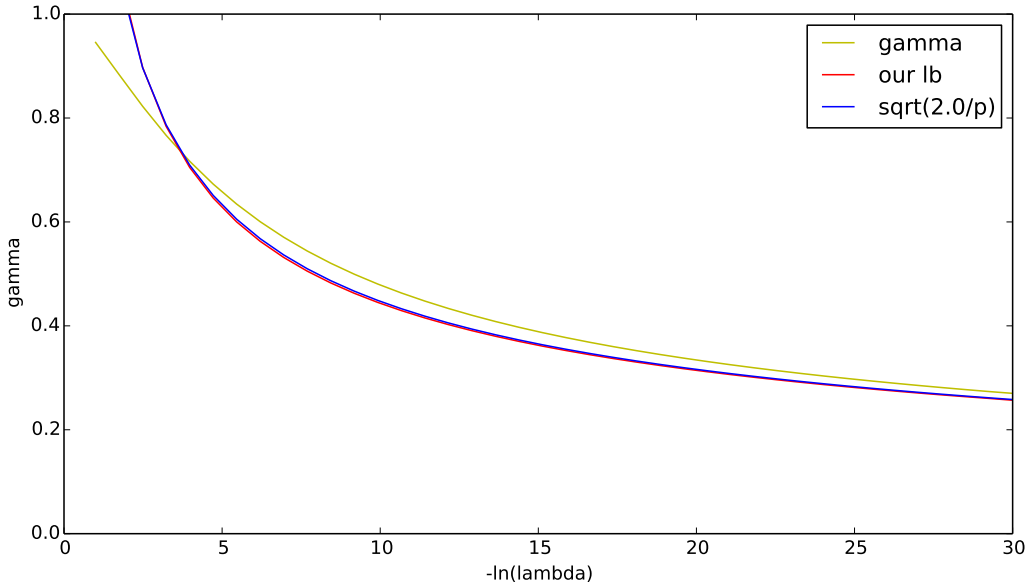

Figure A.2: The behavior of $\min_{0<v<v_*} f_v(v,\lambda)$.

The graph depicts 3 functions: (1) the numerically calculated best $\gamma^*(\lambda)$ given as $\min_{0<v<v_*} f_v(v,\lambda)$ (2) our conjectured lower bound of $\gamma^*(\lambda)$, $\sqrt{2/p}$ (3) a lower bound suggested by our analysis, $\frac{1}{25}\sqrt{\frac{2(p+1)}{p(p-\log p)}}\left[1-\frac{1}{2(p+1)}\right](\hat{v}-\check{v})+\sqrt{2/p}$. These are plotted in terms of $p=-\log\lambda$.

The graph was generated with the "find_gamma_approx.py" script, by running "find_approx_diff()" with options `approx="lam_npow"`, `der_approx="reduced_expr"`, `value_at_approx_estimate = np.sqrt( 2/p )` and `distance_estimate = (vs-v_appr)`.

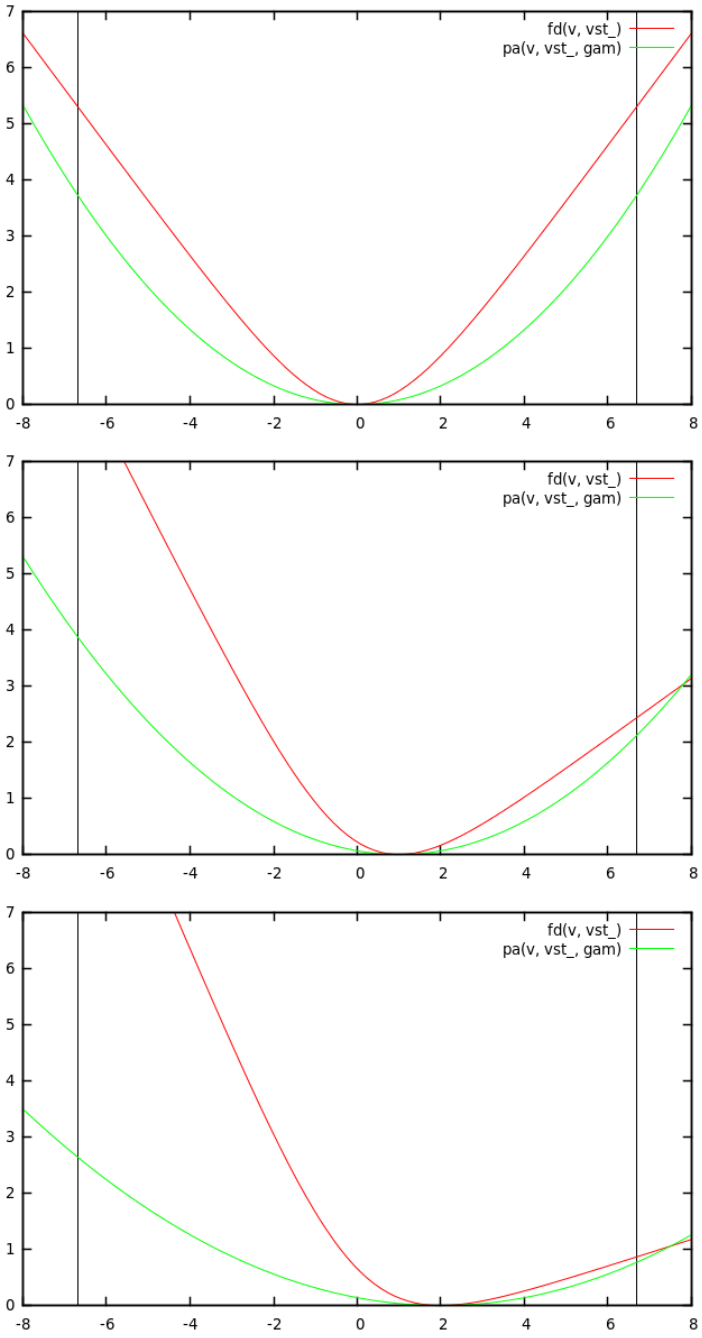

Figure A.3: Illustrating the lower bound l_q^γ with $\gamma(p) = \sqrt{2/p}$ for $v_* \in \{0, 1, 2\}$.

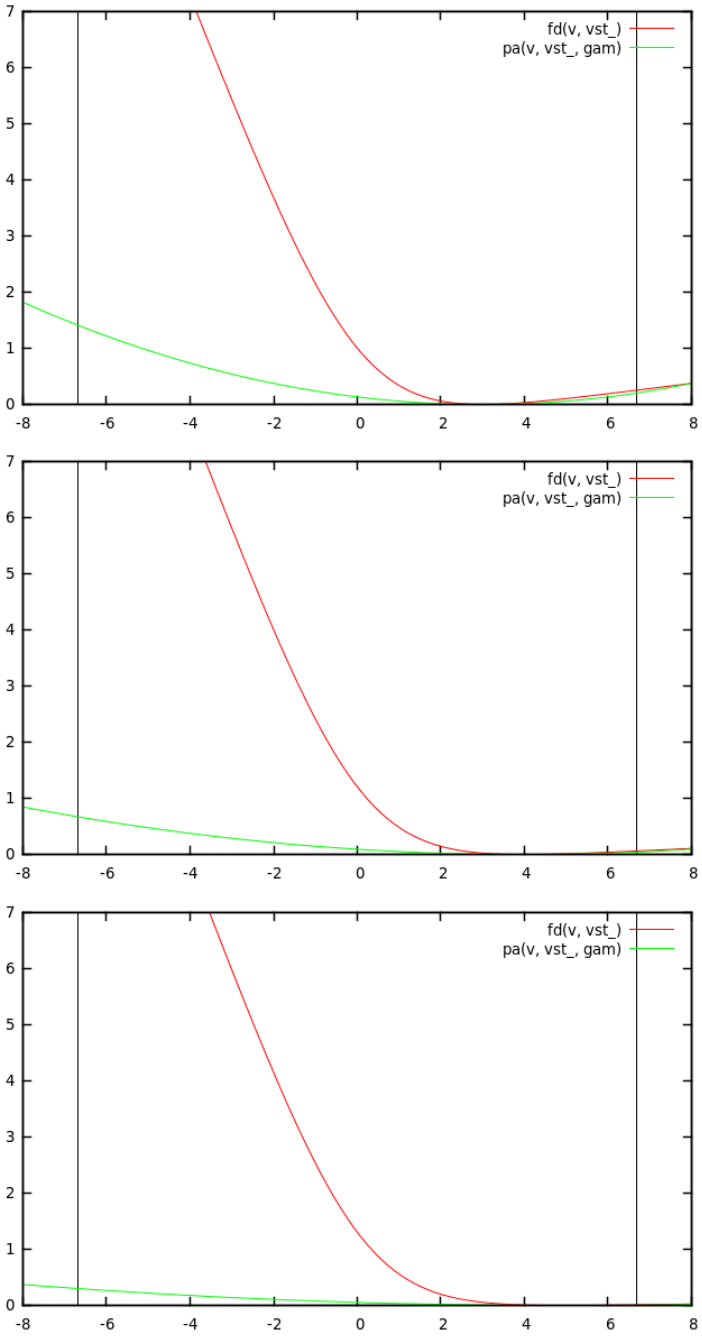

Figure A.4: Illustrating the lower bound l_q^γ with $\gamma(p) = \sqrt{2/p}$ for $v_* \in \{3, 4, 5\}$.

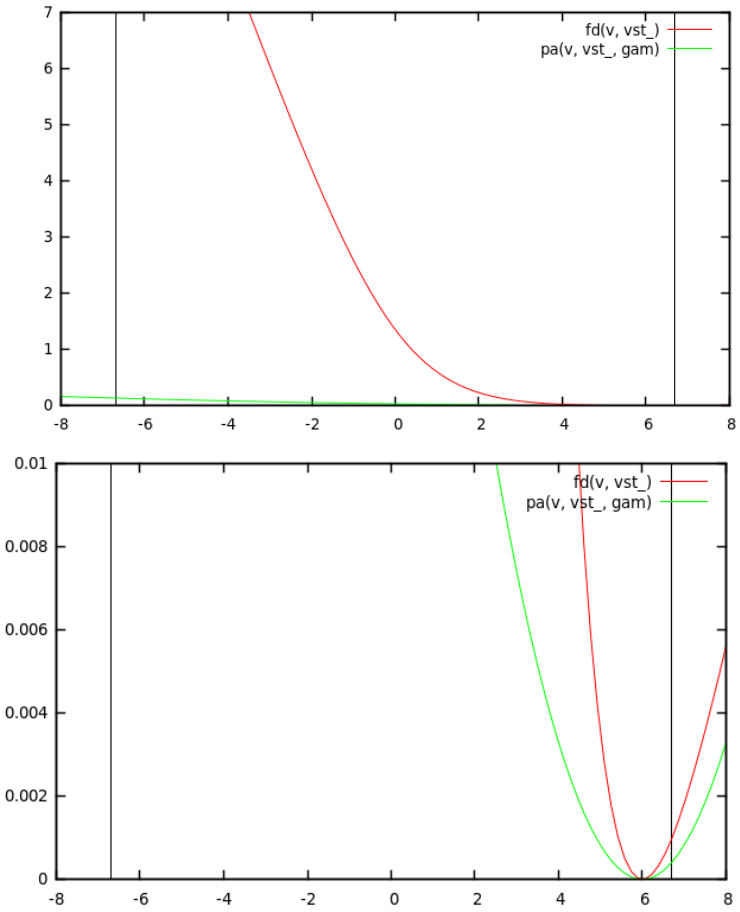

Figure A.5: Illustrating the lower bound l_q^γ with $\gamma(p) = \sqrt{2/p}$ for $v_* = 6$ at the same scale as that of Figures A.3 and A.4, as well as zoomed in.

Appendix B

A Counterexample

We include here the counterexample rendering the result of Gao and Zhou (2013) regarding the global optimizer of the function maximized by EM unproven. We use the notation of the paper for these statements. .

Lemma F.1 relies on inequality (143), which states that for any (p, y)

$$\frac{1}{m}\sum_{j\in S}|y_j - \tilde{y}_j| \leq \sqrt{\frac{1}{m}\sum_{j}|\tilde{y} - y_j^*|}\sqrt{2n\Big(\Delta(p,y) + \Lambda(p,y)\Big) + C}. \tag{B.1}$$

Here we used the notation from the paper, in particular

- $S = \big\{j : |\tilde{y} - y_j^*| \leq \frac{1}{2}\big\}$ (cf page 32 of the supplement),
- $\Delta(p,y) + \Lambda(p,y) = \frac{1}{mn}\Big(\tilde{F}(\tilde{p},\tilde{y}) - \tilde{F}(\tilde{p},y) + \tilde{F}(\tilde{p},\tilde{y}) - \tilde{F}(p,\tilde{y})\Big)$ (cf page 31 of the supplement),
- $C > 0$ is an absolute constant.

We show this claim does not hold. We set $y_j^* = 1$ and all true skills to p^* – by abuse of notation $p^* = p^*\mathbb{1}$. For n such workers these together prescribe an $\tilde{F}_n(p,y)$ function, and in turn its maximizers. The number of tasks is irrelevant to this because it only uniformly scales $\tilde{F}_n(p,y)$ due to the symmetry in the true parameters. From the structure of the maximizers that is described earlier in the paper, and because of this symmetry in the parameters we can WLOG consider maximizers $(\tilde{p}^{(n)}, \tilde{y}^{(n)})$ with $\tilde{y}^{(n)} \geq \frac{1}{2}$ – in fact each

component will be the same. Consequently $S = [m]$, the full task set. We may drop the (n) superscript for convenience at times, when important we will display it, however. We will evaluate the two sides of (B.1) for varying n at $p = \tilde{p}^{(n)}$ and $y = c\mathbb{1}$ for some fixed $c \leq \frac{1}{2}$. We will find that when the number of workers is large enough, the inequality does not hold. The paper defines the quantity $r = \frac{1}{m}\sum_j |\tilde{y} - y_j^*|$, which here is equal to $1 - \tilde{y}$. We will denote by r_n the r corresponding to $\tilde{y}^{(n)}$, and observe that it goes to 0 as $n \to \infty$. We also have $\frac{1}{m}\sum_{j\in S} |y_j - \tilde{y}_j^{(n)}| > c'$ for some absolute constant c'. Before we upper bound the last term in (B.1), we rearrange the objective function $\tilde{F}(p, y)$ to be:

$$\tilde{F}(p, y) = \sum_{i,j} \left[p_i^* \log(1 - p_i) + (1 - p_i^*) \log p_i + y_j(2p_i^* - 1) \log \frac{p_i}{1 - p_i} \right] + \sum_j H(y_j),$$

where $H(\cdot)$ denotes the binary entropy function. We then see for any n, m that

$$2n\Big(\Delta(\tilde{p}, y) + \Lambda(\tilde{p}, y)\Big) = \frac{2n}{mn}\Big(\tilde{F}(\tilde{p}, \tilde{y}) - \tilde{F}(\tilde{p}, y) + \tilde{F}(\tilde{p}, \tilde{y}) - \tilde{F}(\tilde{p}, \tilde{y})\Big) \tag{B.2}$$

$$= \frac{2}{m}\left(\sum_{i,j} \left[\tilde{y}_j(2p_i^* - 1) \log \frac{\tilde{p}_i}{1 - \tilde{p}_i} - y_j(2p_i^* - 1) \log \frac{\tilde{p}_i}{1 - \tilde{p}_i} \right] + \sum_{j=1}^{m} H(\tilde{y}_j) - H(y_j) \right) \tag{B.3}$$

$$\leq \frac{2}{m}\left(\sum_j (\tilde{y}_j - y_j) \sum_i (2p_i^* - 1) \log \frac{\tilde{p}_i}{1 - \tilde{p}_i} \right) + C' \tag{B.4}$$

$$= C' + \frac{2}{m}\sum_j (\tilde{y}_j - y_j) \log(1/r - 1) \tag{B.5}$$

$$\leq C' + 2\log(1/r^{(n)} - 1). \tag{B.6}$$

Going from (B.2) to (B.3) we simply cancelled equal quantities. In the next line we collect terms and use that the difference of the entropies is bounded, as the entropy function is bounded from both below and above. Step (B.4) to (B.5) uses equality (44) from the paper, and for the final step we note $\log(1/r - 1)$ is non-negative as $r \leq \frac{1}{2}$ so we can replace $\tilde{y}_j - y_j$ by the larger 1 for an upper bound. Additionally, we make explicit that the r in the inequality is that which corresponds to having n workers. In conclusion, for any n, m we have

$$c' \leq LHS \overset{?}{\leq} RHS \leq \sqrt{r^{(n)}}\sqrt{2\log(1/r^{(n)} - 1) + C''} \tag{B.7}$$

This cannot be true for every n, however, as the rightmost expression goes to 0 when $n \to \infty$:

$$\lim_{r^{(n)} \searrow 0} r^{(n)} \left(2\log(1/r^{(n)} - 1) + C''\right) = \lim_{x \to +\infty} \frac{2\log(x - 1)}{x} + 0 = 0.$$

Since there are certain assumptions on the parameters where this lemma is eventually applied, we make sure those are also met in our example. Namely, Lemma F.1 is used in the proof of Lemma 6.3, in "case E.1" where $\frac{1}{m}\sum_j \hat{r}_j(1-\hat{r}_j) \geq \exp(-\beta n)$ for β defined as $\log \frac{1}{\min(r,1-r)}$. Here $\hat{r}_j$ is $\frac{1}{m}\sum_j |y_j - y_j^*|$ applied at certain $\hat{y}$. So we use c and get $LHS = c(1-c)$, $RHS = [r^{(n)}]^n$. Since $r^{(n)}$ already tends to 0 with increasing n, so raising it to the n^{th} power only makes this faster and the inequality will be satisfied for large enough n.